Sim3D_MaxGUI™

User's Guide

The Graphical User Interface

for

Sim3D_Max™

A.V. Cazan
10/10/2004

1. INSTALLING SIM3D_MAXGUI 3
2. GENERAL DESCRIPTION 4
3. THE MAIN VIEWS 5
3.1. THE TREEVIEW 5
3.1.1. Editing the values 7
3.1.2. Units of measure 8
3.1.3. Multiple values in same line 9
3.1.4. Special cases 10
3.1.5. Copy and Paste 11
3.2. THE GRAPHICVIEW 12
3.2.1. Viewing the geometry data 12
3.2.2. Viewing the Spatial Grid 18
3.2.3. Editing the geometry data 18
3.2.4. Material Color Pool-file 21
3.3. THE LOGVIEW 26
4. THE DATA VIEWS 26
4.1. MATERIAL INDEX VIEW 28
4.2. FIELD OUTPUT VIEW 30
4.3. TIME PROFILE VIEW 33
4.4. SOURCE VIEWS 34
4.4.1. User Source 34
4.4.2. DIFFRACT Source 34
4.5. EXPORT REFLECTED VIEW 36
4.6. EXPORT TRANSMITTED VIEW 37
4.7. MONITOR VIEW 38
5. THE DATA 3D VIEWS 42
5.1. COMMON FEATURES 42
5.1.1. Orientation and Zoom 43
5.1.2. Color Legend 44
5.2. SOLID SCENE 45
5.2.1. Loading and preparing the data 45
5.2.2. Scene Options 46
5.2.3. Lights 51
5.3. RELIEF SCENE 52
5.3.1. Scene parameters 53
5.3.2. Margins 54
5.3.3. Z-Value picker 55
6. FILES AND DIRECTORIES 56
7. LAUNCHING SIM3D_MAX 57
8. THREE WORDS 58

1. Installing Sim3D_MaxGUI

The Sim3D_MaxGUI package consists of only one executable, Sim3D_MaxGUI.exe. Hence, there is no need for complicated installation procedures: simply copy the executable file into any directory of your choice. Note that this directory must not be correlated in any way to the directory where Sim3D_Max is installed. Please read the chapter **Files and Directories** in this manual to find out more about how Sim3D_MaxGUI deals with the various required files and folders.

For running, Sim3D_MaxGUI does not need any additional files; it can be started like any other program under the Windows operating system.

To uninstall Sim3D_MaxGUI, simply delete the executable file from your file system.

For a better understanding, it is recommended to read first the Sim3D_Max manual before starting with this guide.

2. General Description

Sim3D_MaxGUI has been developed as an interfacing tool for the Sim3D_Max program. As such, it can deal with the same files as Sim3D_Max does. It can be regarded as a kind of control center for creating and editing the various input files of Sim3D_Max, for viewing the content of the Sim3D_Max output files and for launching the Sim3D_Max program.

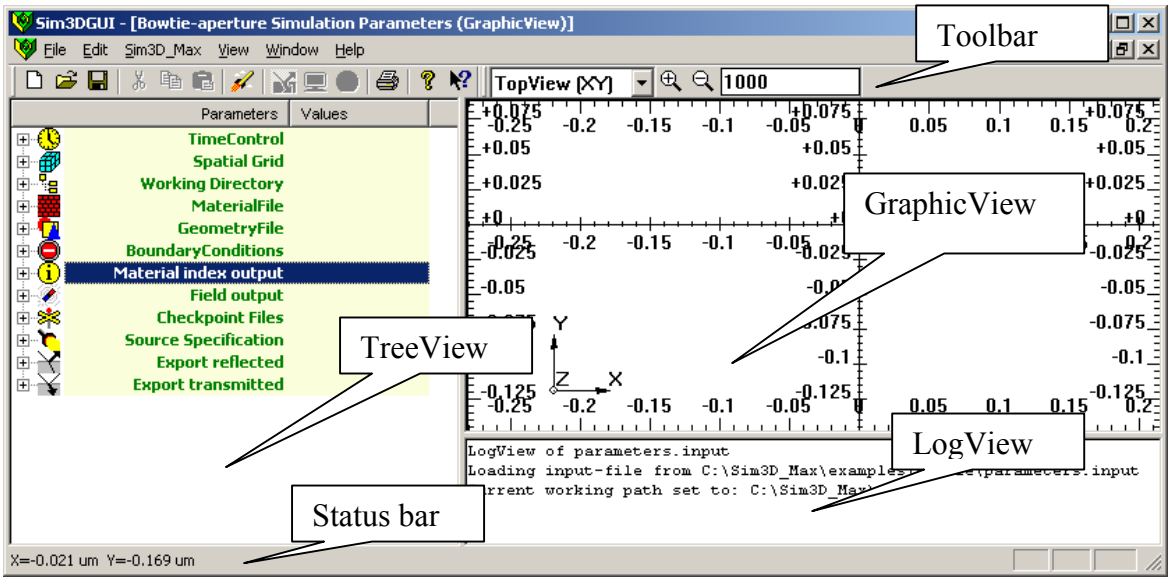
Among the many different files read by Sim3D_Max, the *parameters.input* files play a central role. Therefore, Sim3D_MaxGUI uses the same file type as a base for further working.

The following picture displays the general aspect of Sim3D_MaxGUI, after loading such a *parameters.input* file. More then one parameters.input may be open at same time. For each of them, a main window is spawned, containing three panes (called Main Views):

- TreeView
- GraphicView
- LogView

At a given time only one of the three pane-windows has the input focus (being activated), which means it can receive mouse and keyboard input. Depending on which pane has the input focus, some controls of the toolbar will be disabled or enabled. A pane will be activated automatically when a mouse click is performed on its area. To avoid confusion, the actual active pane is indicated with the text TreeView, GraphicView or LogView in the caption of the main window.

The following chapter will describe the functionality and the usage of each of these panes.

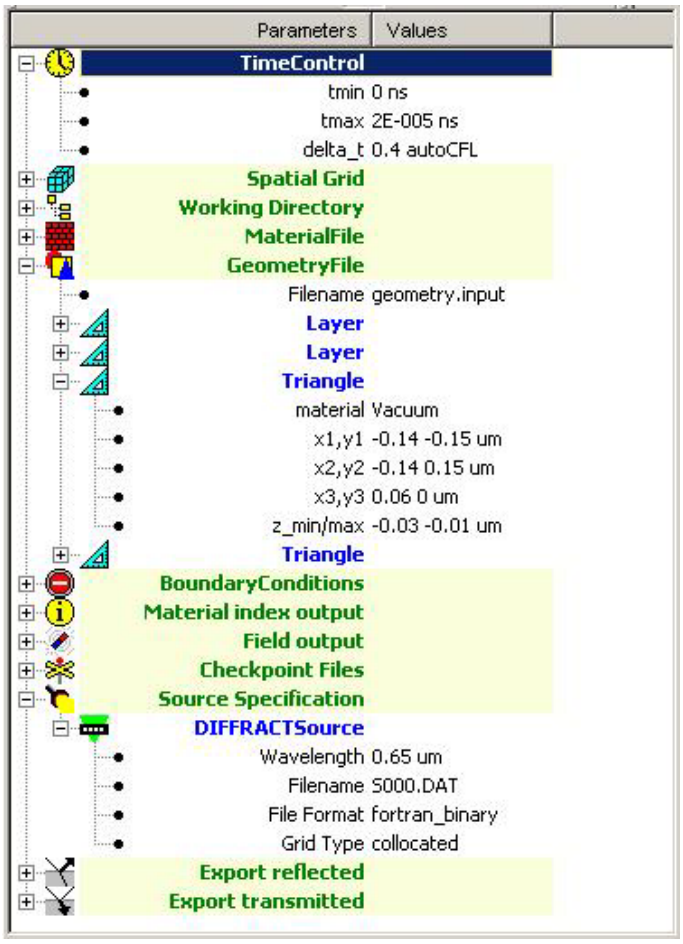


3. The Main Views

3.1. The TreeView

The TreeView is the most important of the panes. It displays the data of the loaded *parameters.input* file in an editable, structured tree-list. As an example, see the picture below.

The parameters are grouped by similar rules like those described in the User Manual of Sim3D_Max. Each group can be expanded or collapsed arbitrarily, to make visible or to hide the underlying parameters. This may be done with a click on the +-nodes or by opening the popup-menu with a right mouse button and selecting the **Expand** or **Collapse** command.



For each entry of the tree there is a corresponding popup menu which will be open upon a right mouse click on that entry. Depending on the selected tree entry, one or more of the following possible command options may appear in the popup menu:

- **Expand tree** expands the selected tree node if it is not already expanded, otherwise it has no effect.

- **Collapse tree** collapses the selected tree node back if it is not already collapsed, otherwise it has no effect.

- **Edit** opens an edit- or a combo-box, allowing to change in-place the value of the selected parameter. The same effect may be achieved by a mouse double click on the value-string of the parameter.

- **View** opens one of the available DataViews (see next chapters for a more detailed description). Which DataView is opened depends on the selected tree entry. There are two exceptions from this rule. The first exception is for geometry objects: when selecting View for such an tree entry, then no DataView will be handled, but the corresponding geometry object will be sketched in (or removed from) the GraphicView (read the GraphicView chapter for more information). The second exception is for the **SpatialGrid**: as in the case of geometry objects, the computational grid will be drawn in (or erased from) the GraphicView.

- **Main cell only** is an option that appears only for menus attached to Lattice or Pattern geometric primitives. When selected, only the main cell (the cell having the indices 0,0) of the Lattice or Pattern will be displayed. Since the other cells are only repetitions of this, it is generally sufficient to view only the main cell. In the case of large-structured objects this option may spare lot of time during the display.

- **Browse …** This option will appear only in the popup menus attached to filename parameters. It offers the possibility to assign a filename without using the keyboard. However, this is only useful if the filename represents an existing file.

- **Remove** Some of the command blocks specified in *parameters.input* do not have a fixed length, but contain instead an arbitrary number of subitems. For example, multiple **ExportTransmitted** blocks are allowed to be present. Another example might be the variable number of entries in the **MaterialFile** or the **GeometryFile** blocks. Such existing entries can be deleted with the **Remove** command.

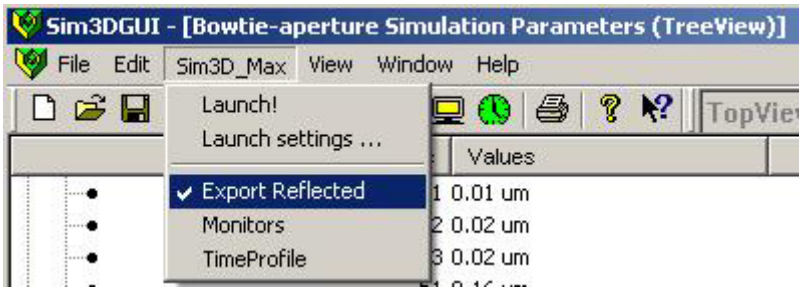
- **Insert new** always appears in conjunction with **Remove**, allowing the user to insert new entries in the situations mentioned above.

- **Load … file** loads the content of the specified file into the TreeView. Typically, this command will appear for the **MaterialFile, GeometryFile,** and the BoundaryConditions tree entries. Note that when a *parameters.input* file is loaded into Sim3D_MaxGUI, the *material.input, geometry.input,* and *boundaries.input* files specified in the loaded *parameters.input* file are NOT loaded automatically into the TreeView.

- **Save … file** Appears always in conjunction with **Load … file**, but serves to save back the data into the specified *material.input* or *geometry.input* files. Notice that when a parameters.input file is saved from Sim3D_MaxGUI to disk, then the *material.input* and *geometry.input* files specified in the stored *parameters.input* file are NOT saved automatically. Use this menu command to perform this task.

- **Append new …** This commands has the same effect like **Insert new**, but instead of inserting a new entry immediate after the selected tree entry it appends the new entry at the end of the entry list of items. Hence, it is the same like applying **Insert new** on the last element of an existing list. It is especially useful when the edited list is empty, since when the edited list is empty there is no access to the **Insert new** command.

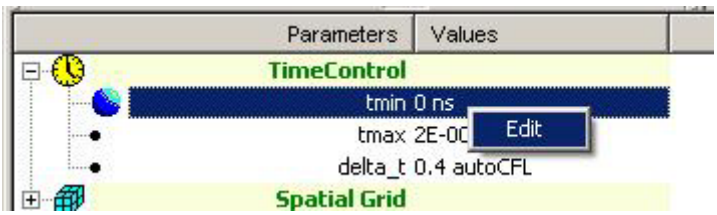
Some of the command block entries of the *parameters.input* file are optional (**ExportReflected**, **ExportTransmitted**, **Monitor** and **TimeProfile**).



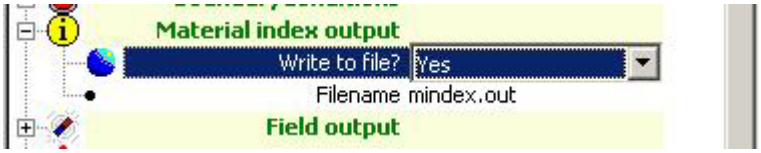
With the above displayed main menu commands ***Sim3D_Max-Export Reflected, Sim3D_Max-Monitors*** and ***Sim3D_Max-TimeProfile*** it is possible to activate or deactivate those entries from the TreeView (and consequently from the parameters.input file under consideration). The toolbar contains three buttons as a shortcut to these menu commands ( ). Note that a menu command **Export Transmitted** is not present. This is not necessary because, according to the rules for the *parameters.input* files stated in the Sim3D_Max manual, it is not allowed to place an **ExportTrasmitted** block without preceding **ExportReflected** block. Furthermore, if an **ExportReflected** block is present, then at least one **ExportTransmitted** must follow.
Upon activating the **ExportTransmitted** or **Monitor** block, any arbitrary number of **ExportTransmitted** or **Monitor** blocks may be added to the existing ones.
Finally, it should be emphasized that after performing changes in TreeView, the ***File-Save*** or ***File-Save As*** menu command must be used to save this changes into the desired *parameters.input* file on the hard disk.

3.1.1. Editing the values

It is very easy to change the numerical values of the various parameters displayed in the TreeView. There are three ways to start the editing operation. Perform a right-button mouse click on the desired parameter. From the popup menu select ***Edit***. Note that in many cases the popup menu will have only this command, as shown in the following picture:



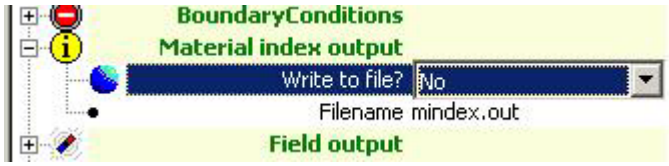
Alternatively, a double-click with the left mouse button on the value column (that is '0 ns' in the above example) would lead to the same result: the opening of a small edit- or combo-box, depending on the type of parameter, like in the following pictures:



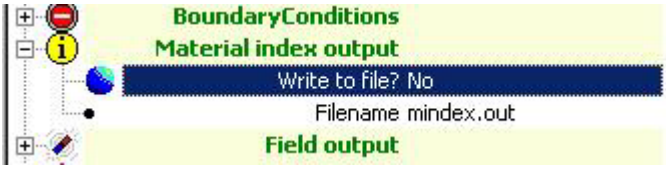
The third, and simplest way to start editing is to press simply the ENTER key.

When a combo box is opened, first expand the box's list with a click on the button of the box:

Then select the desired option, let's say 'No' for this example:



Finally, click with the mouse elsewhere on the screen (outside the combobox) or press ENTER, and the changes will be taken into the TreeView:

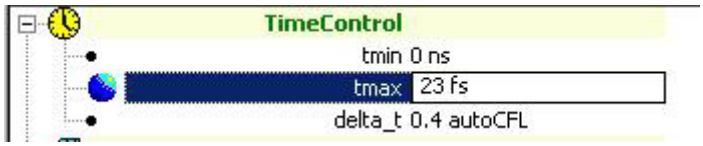


An edit box can be closed in the same way, but for the edit box input, some further pointsare made in the text that follows.

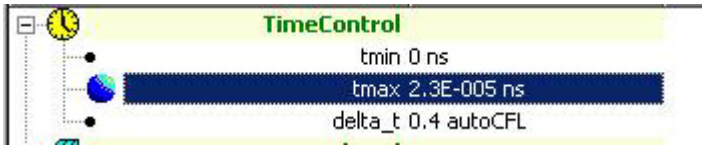
3.1.2. Units of measure

The values of many parameters are given in specific units of measure. Lengths are specified usually in microns (um) and time values in nanoseconds (ns). To find out what unit is required for a given parameter, please refer to the Sim3D_Max manual.

Sim3D_MaxGUI will display in the TreeView the values always measured with the unit expected by Sim3D_Max, but during input other units may be specified in the edit box. If such a non-standard unit is used, the accompanying value will be converted by Sim3D_MaxGUI to the unit expected by Sim3D_Max for that parameter. For example, you may specify:



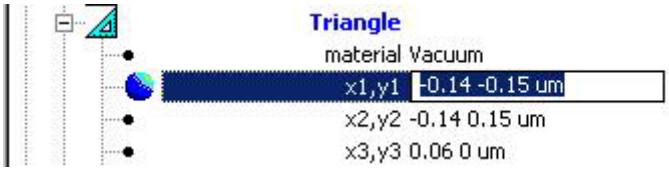
After closing the edit box, Sim3D_MaxGUI convert the femtoseconds (fs) into the required nanoseconds (ns):



The same behaviour is valid for other physical values like lengths or frequencies. Optionally, the unit may be omitted from the input, in which case Sim3D_MaxGUI assumes the value to be already specified in the units expected by Sim3D_Max and hence it performs no further post-edit conversion.

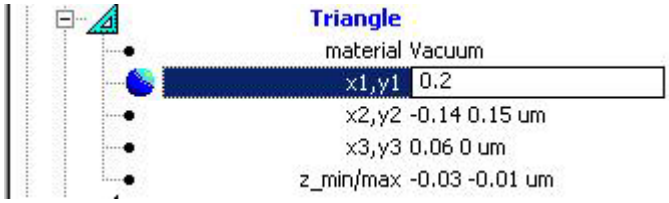
3.1.3. Multiple values in same line

A somewhat special situation arises when entries are edited that have multiple values in the same line, such as:



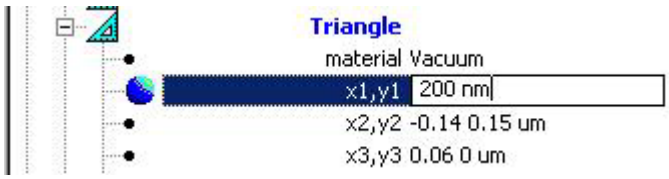
It should be noted that all values on the same line must have the same units. After closing the edit box, Sim3D_MaxGUI splits the line into sub-strings (tokens, separated by blanks) and expects one token for each value of the line, plus an extra token for the unit of measure. However, only the first token is mandatory, the others are optional. If they are missed, the corresponding value is set identical to the preceding value.

As an example, suppose we want to set both the x1,y1 values to the same value, say 0.2 um. To achieve this, we must type:

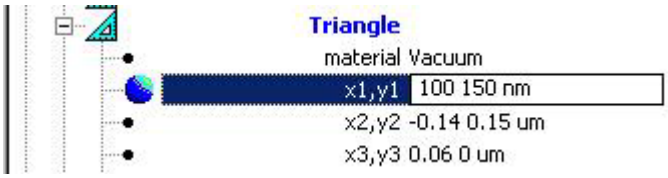


The result would be:

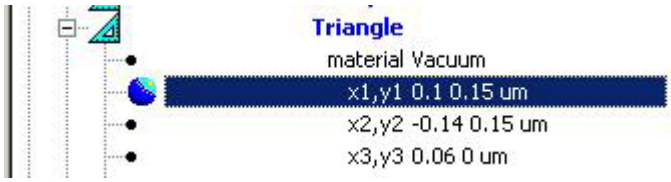
With the same result we could have typed:



To set the values to individually, different values, type: (note the obligatory spaces between the values)



With the result:



Of course, the unit mnemonic (**nm** in the above input example) can be omitted, in which case the input will be regarded as microns and set to x1,y1=100,150 um.

Use the Esc key to abort the edit operation and to restore the original values before editing.

3.1.4. Special cases

Unfortunately not all parameters can be handled with the mechanisms described above. However, there are only a few exceptions from the rule. In the following text, all parameters requiring special or additional handling are explained.

The first exception is the ***delta_t*** parameter of the **TimeControl**-block. According to the Sim3D_Max manual, the ***delta_t*** must be specified in nanoseconds or dimensionless, by specifying the keyword *automatic-with-CFL* in the *parameters.input* file. In Sim3D_MaxGUI this can be achieved by specifying either **ns** or **autoCFL** (for the second case).

The second exception is the ***t_on,t_off*** parameter line of the **Monitor**-block. You may specify something like **10e-6 20e-6 ns** or something like **10 periods** (or more simple: **10 per**) to distinguish between the two cases stipulated in the Sim3D_Max manual (see there for details).

Two other slightly nonconforming lines are found in the **FieldOutput**-block: the ***WriteEx,Ey,Ez?*** and ***WriteHx,Hy,Hz?*** lines. Sim3D_Max expects to find three Yes/No options in these lines, instead of the usual one per line. When the user requests editing for these lines, Sim3D_MaxGUI opens a single edit box, instead of three combo-boxes. Type the desired sequence of Yes/No words into this box. For convenience, Sim3D_MaxGUI will accept the letters **y** or **n** in place of explicit **yes** or **no** entries. As an example, the sequence **y n n** will be recognized as **Yes No No**. Note the mandatory spaces between the letters y and n. The same result can be obtained by typing only **y n**. If you want all three options to be the same, then only the first must be specified. Example: **y** will be translated to **Yes Yes Yes** and **n** to **No No No**.

Inside the **ExportReflected** and **ExportTransmitted** blocks, the line containing the parameters ***NX,NY*** also needs special handling. In that line Sim3D_Max expects a pair of integers (usually both a power of 2), but these lines can be omitted from the *parameters.input*. If you want to prevent the line to be written into the saved *parameters.input*, specify 0 as value. In this case, for convenience, Sim3D_MaxGUI displays the keyword **As_In_Source** instead of 0, indicating that this line will not be present in the *parameters.input* file.

3.1.5. Copy and Paste

These menu commands allow to copy or paste whole TreeView entries to or from the Windows clipboard. The data is stored as plain text blocks, hence this functionality may be used to transfer entries between Sim3D_MaxGUI and any ordinary text editor.

The most powerful usage of the copy and paste commands is the capability to transfer blocks between different *parameters.input*, *geometry.input* or *material.input* files.

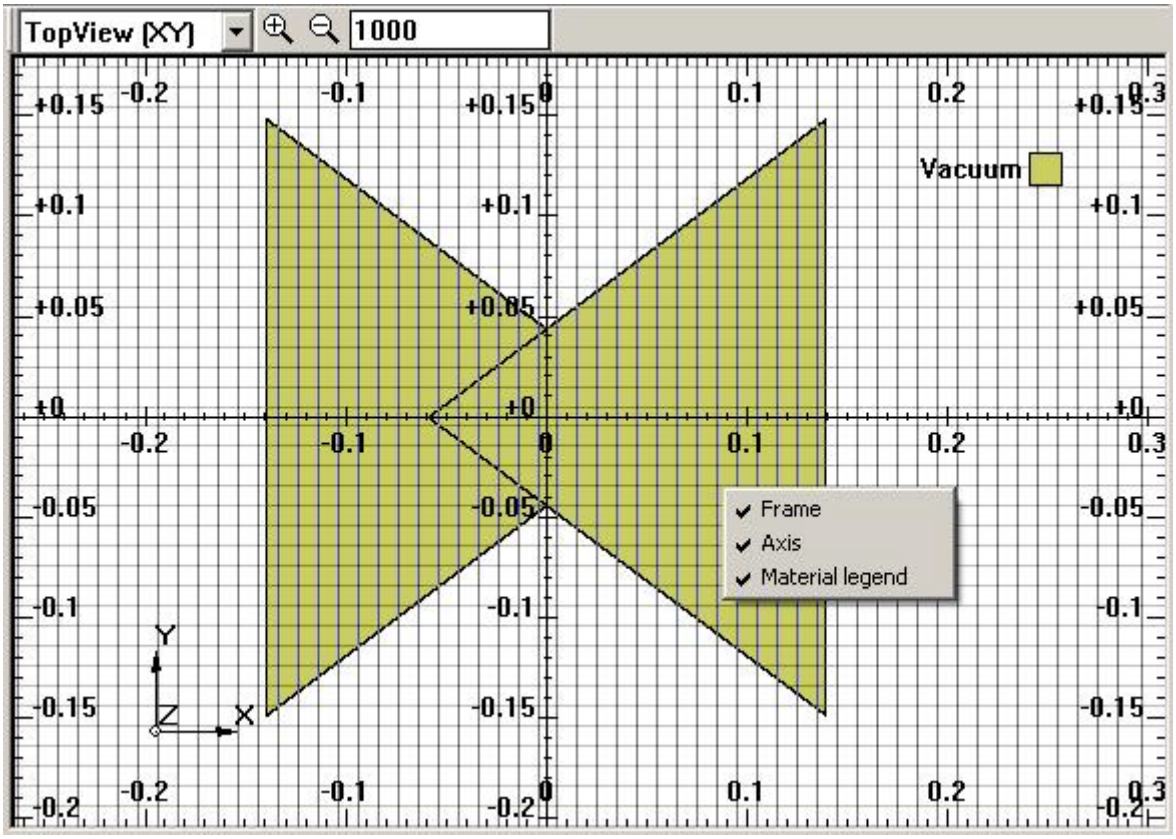
When copying to the clipboard, the TreeView places two extra lines containing keywords at the beginning of the text block. These extra lines are used to identify the text block for the following paste operation.

As a general rule, when pasting, TreeView overwrites the existing tree entry. For example, according to the syntax rules of the parameters.input files, only one SpatialGrid entry is allowed. Hence when a SpatialGrid block is pasted from clipboard, the existing one will be overwritten. An exception is when a geometric primitive is pasted, the new primitive is inserted immediately after the highlighted geometry from the existing geometries. This allows an easy duplication of existing graphical primitives within the geometry data, which afterwards may be modified by any method described in this guide.

3.2. The GraphicView

The GraphicView is designated to display and edit the content of the *geometry.input* file. Additionally, if desired, the computational grid lines will be drawn inside this view.

3.2.1. Viewing the geometry data



The popup menu (opened with a right mouse-button click) offers the possibility to decorate the view with a coordinate frame, coordinate axes and/or a color legend for the materials of the displayed objects. By default, all three options are activated, as shown in the above picture.
The units of the coordinates are always microns.

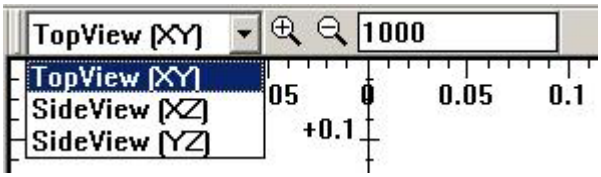
The GraphicView offers three ways to display the geometry objects defined in the **GeometryFile** parameter block of the TreeView:

- **TopView (XY)**. The objects are projected on the XY-plane and the observer is located on top of the Z-axis, looking toward the axis.

- **SideView (XZ)**. The objects are projected on the XZ-plane and the observer is located on top of the Y-axis, looking toward the axis.

- **SideView (YZ)**. The objects are projected on the YZ-plane and the observer is located on top of the X-axis, looking toward the axis.

The desired mode can be chosen from the combo-box of the toolbar:



Note that an orthographic projection is used, that means no depth information is considered (the far objects have same size as the near objects). The geometry objects are drawn in the same order as they appear in the **GeometryFile** entry of the TreeView. As a consequence, later-defined objects may obscure some of the previously-defined objects which were drawn first on the screen.

The number contained in the editbox of the toolbar (in the above picture this number is 1000, which is also the default value) represents the number of pixels per micrometer that will be used to scale the objects on the screen. It may be changed by using the buttons on the toolbar (this changes the value in 10% steps) or by typing any arbitrarily value greater then zero into the editbox and pressing ENTER. Note that the ENTER key is important; otherwise Sim3D_MaxGUI will not regard the number from the editbox as a valid number and will overwrite it with the old value at the next screen redraw. As a simple and convenient alternative, the scale factor may be changed by rotating the scroll wheel of the mouse device attached to the computer. However, to use this facility, the GraphicView must have the input focus, otherwise the scroll wheel messages will be redirected to that window (or control element) that has the input focus. To give GraphicView the input focus, perform a left button mouse click somewhere on the view's area.

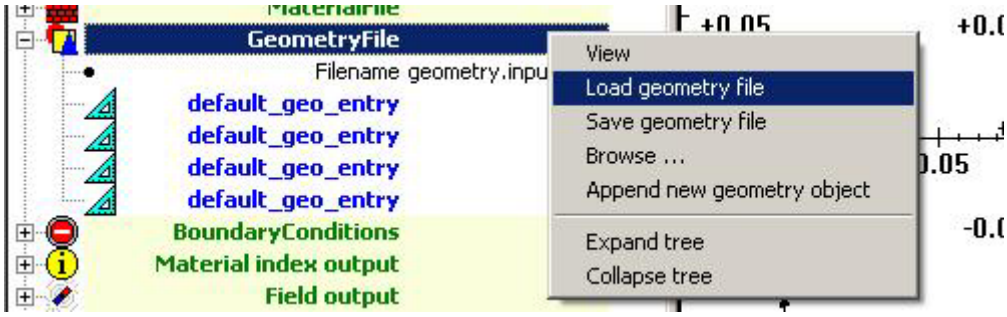
To translate (pan) the drawing on the screen press the left mouse button down, hold it down and move the mouse cursor and then release the left mouse button. When the left button is released, the GraphicView will repaint the geometric objects shifted by the same amount as the previous displacement of the mouse cursor

The color used to fill the geometric shapes is determined by character string of the material name, which means different material names will usually lead to different colors. However, if the material names differ only slightly, the colors of different materials may appear identical on the screen.

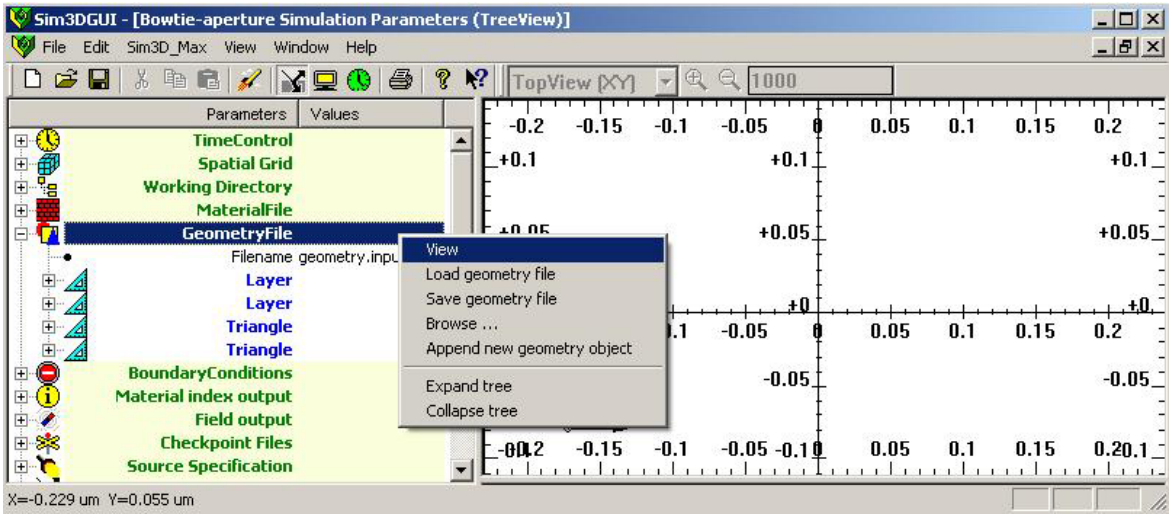
When moving with the mouse cursor inside the GraphicView, the cursor coordinates and the material name (if the cursor is over a defined material) will be displayed on the left side of the status bar, which is on the bottom of the screen.
A double-click with the left mouse button on any shape inside the GraphicView will expand or collapse the tree node in the TreeView that corresponds to the geometric object under the mouse cursor. This is a quick and convenient way to get access to the numerical parameters of the displayed geometric objects.

It is important to notice that when a *parameters.input* file is loaded into Sim3D_MaxGUI, then the *geometry.input* file specified in the *parameters.input* is NOT loaded automatically.

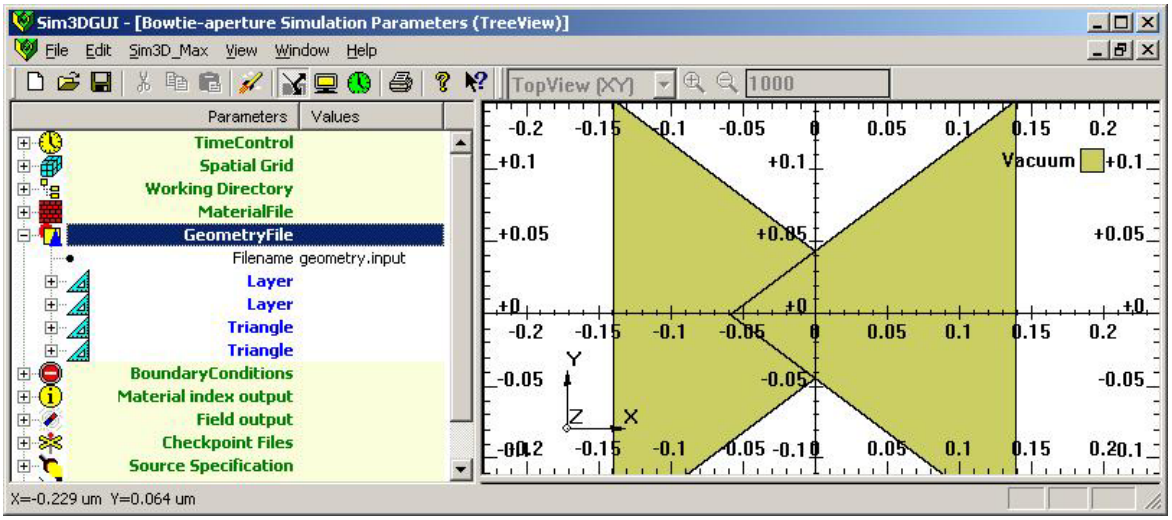
To read the *geometry.input* file open the popup menu of the **GeometryFile** entry of the TreeView and select the command ***Load geometry file***, as shown in the picture below:



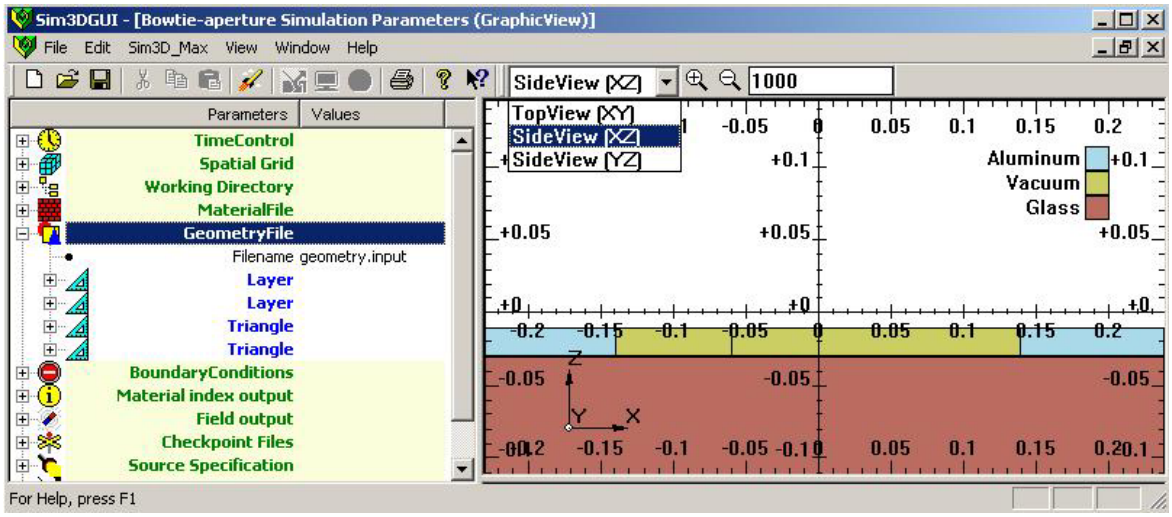
Initially, the GeometryFile block has only some default placeholders for geometry objects. After the geometry.input file has been loaded successfully, the TreeView may look like:



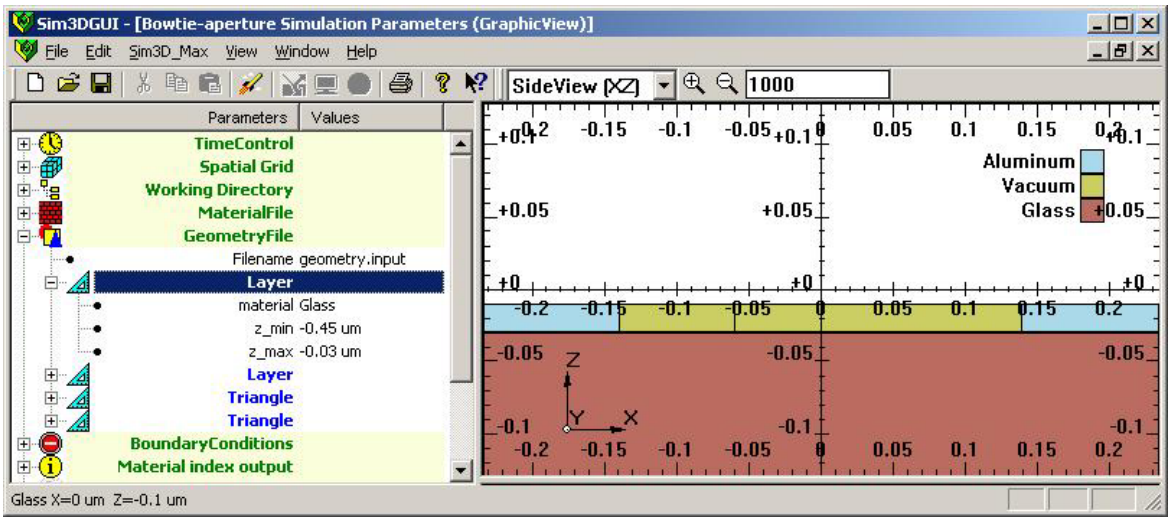
Now select the ***View*** command of the popup menu to instruct Sim3D_MaxGUI to display the **GeometryFile** entries in the GraphicView. Notice that you must open the popup menu of either the **GeometryFile** or the **Filename** entry. No other popup menu will do this. The result may look like:



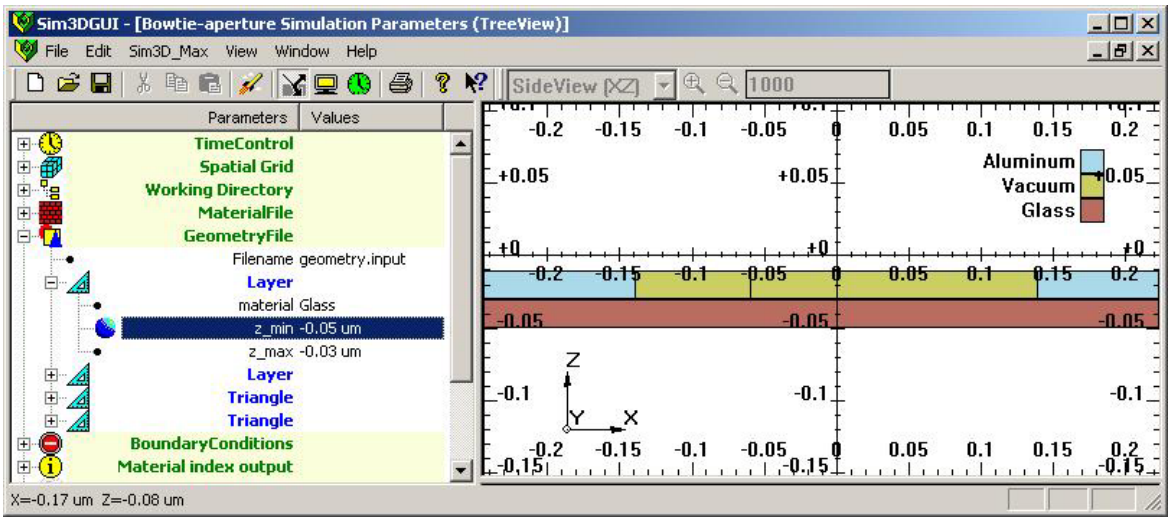
First, notice that the objects are drawn in the same order as they appear in the TreeView. In the above example, the first Triangle is partially obscured by the second Triangle.
Second, not all geometric primitives are drawn in every display mode of the GraphicView. For example, in the above picture the first two *Layer* entries are not drawn in this mode. Please refer to Appendix A to find a complete list of examples that describe how the geometric primitives are drawn in each mode.
To cause the GraphicView to show the *Layer*s too, one must change the display mode to a SideView:



Now move the mouse cursor near the coordinates X=0, Z=-0.1 and perform a double click with the left mouse button. As a result, the first Layer entry of the TreeView will be expanded as shown below:

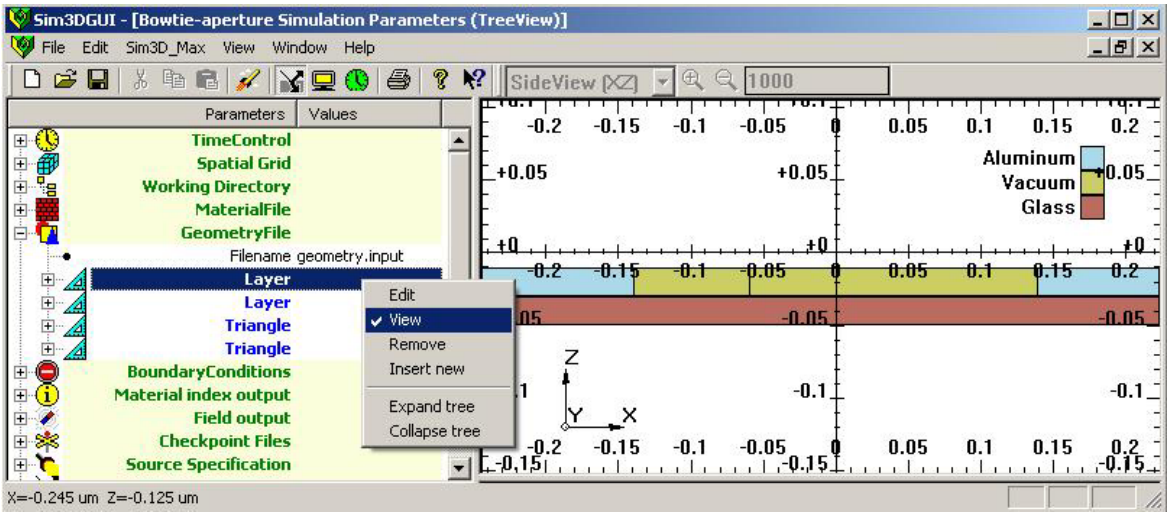


In the above example, the result would be the same provided that the mouse cursor was anywhere in the area of the first Layer, that is any arbitrarily X coordinate and any Z coordinate less then –0.03 microns.
The GraphicView will immediately reflect any changes that are done to the numerical parameters of the geometry primitives. If, for example, you change the z_min parameter of the first Layer from –0.45 microns to –0.05 microns the GraphicView would display:

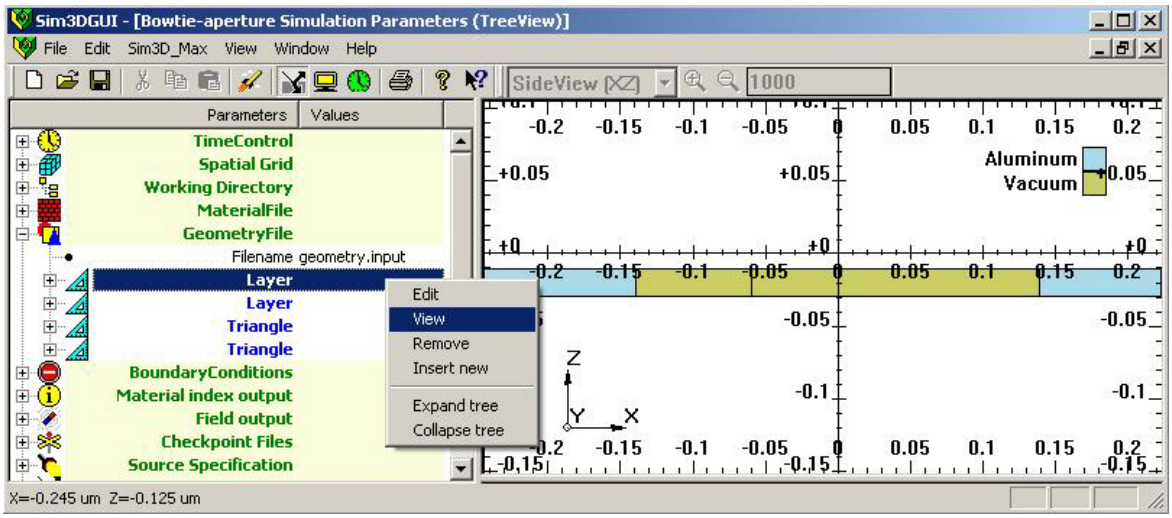


After finishing editing the expanded Layer, it is recommended to collapse the tree entry back, to avoid a crowded TreeView. Any method is acceptable to do this, but in the above situation the fastest way is to perform this is by performing another double-click on the area of the Layer drawing inside the GraphicView.

Every geometric primitive displayed as an entry in the TreeView has attached a popup menu containing, among other commands, a ***View*** command. Invoking this command will instruct the Sim3D_MaxGUI to detach (or attach) the geometric primitive from (to) the list of displayed geometric shapes. If the geometric primitive is attached to the GraphicView (that is, when it is enabled to be displayed), then the popup menu ***View*** will appear marked with a check-sign, as shown below:



In the above situation, when the ***View*** command is invoked, the first Layer will be removed from the GraphicView:



Notice that the Glass entry has disappeared from the material legend of the GraphicView, together with the Layer. The material legend contains only the materials of those objects that are really drawn on the GraphicView, but since the GraphicView displays only a clipped rectangular area, not all objects are necessarily visible. Hence, if the material legend contains a material that is not visible on the screen, this indicates the existence of other objects that cannot be seen for one of two possible reasons. Either they lie outside the visible area or they are being obscured by other geometries that occupy the same space and were drawn first or are equal or larger in size to the obscured geometry and have a greater value for the third coordinate (the one not included in the current view) than that of the obscured geometry.
For the case where the geometry lies outside the viewing area one should zoom or pan the visible area to find out where those objects lie. For the other cases one should either reverse the order of the geometries in the TreeView or change the view to one of the two other view choices.

To restore the initial display, simply use again the ***View*** command of the Layer-popup menu.

The ability to attach/detach each geometric primitive from the GraphicView is very useful when objects are obstructing each other on the screen: the objects on the top may then be detached one by one, revealing the objects underneath. Of course, this feature works only if the complete GeometryFile is enabled for drawing (that is, when the popup menu of the GeometryFile or Filename entry is selected). Otherwise, selecting the View command from the popup menu of any individual geometric primitive will remain meaningless, without any effect.

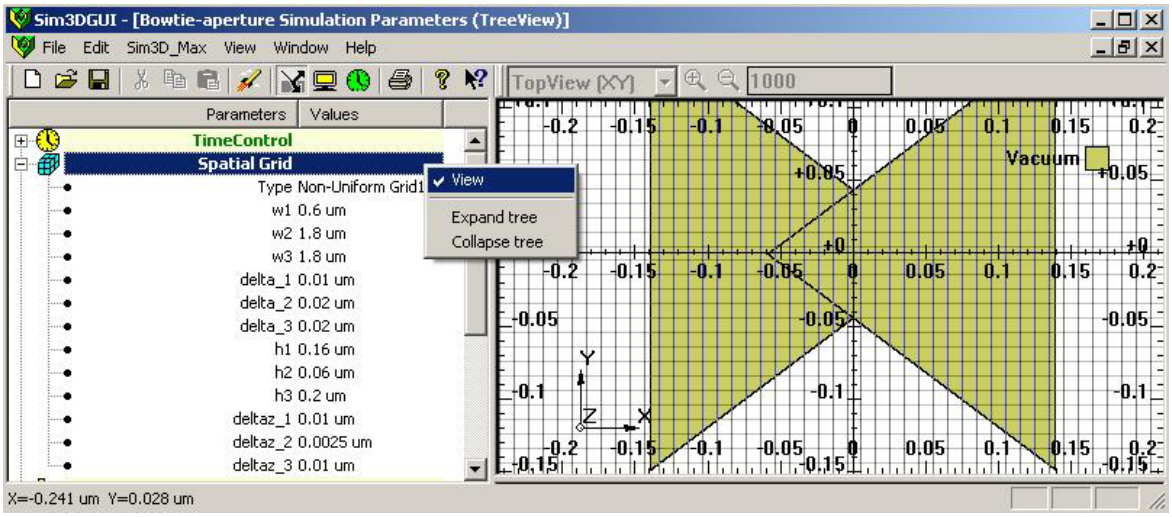
Some of the geometric primitives require more time for drawing then other. Especially filigreed lattices which can take several seconds to get fully painted. If you want to remove those objects from the screen, to avoid long pauses, detach them with the above method from the GraphicView.

In the case of a Lattice or a Pattern, the popup menu offers an additional command option named ***Main Cell Only***. When this option is activated (default), then only the content of the main cell of the corresponding lattice or pattern is displayed. This greatly reduces the amount of time it takes to redraw the view when the user zooms or pans. For reference purposes, the centers and corners of all other cells of the lattice or pattern will be indicated by red crosses on the screen.

When the whole GeometryFile is attached to the GraphicView (that is, when the View command of the popup menu of the GeometryFile or Filename entry is selected), all geometric primitives will automatically become attached, regardless of their previous View-flags. This functionality may be used in the case of a GeometryFile with many geometric primitives, where an unknown, confusing number of them have been detached from the GraphicView. To avoid opening the popup menus for each of the primitives: first detach the GeometricFile and then attach it back. By doing so, one has the certainty that all primitives are drawn in the GraphicView.

3.2.2. Viewing the Spatial Grid

In addition to the geometric primitives, the computational domain defining the ***SpatialGrid*** may be displayed in the coordinate system of the GraphicView. To attach the ***SpatialGrid*** to the drawing, select the ***View*** command from the popup menu of the ***SpatialGrid*** tree entry, as indicated in the picture below:



This feature is useful to determine if the specified computational domain matches the defined geometry.

As in case of the geometric primitives, the GraphicView will immediately reflect any changes that are performed to the SpatialGrid numerical parameters.

3.2.3. Editing the geometry data

Despite to its name, the GraphicView also offers the possibility to modify graphically the various parameters of the displayed primitive geometric objects. In contrast to the TreeView, where the parameters may be modified only with the keyboard, within the GraphicView the mouse can be used to modify the shape and orientation or to drag the objects around on the screen. However, due to the limited resolution of the screen, some loss in precision has to be paid for this comfort. In order to precisely determine the material within each cell of the computational grid it is recommended that the user should generate a material index file and examine it.

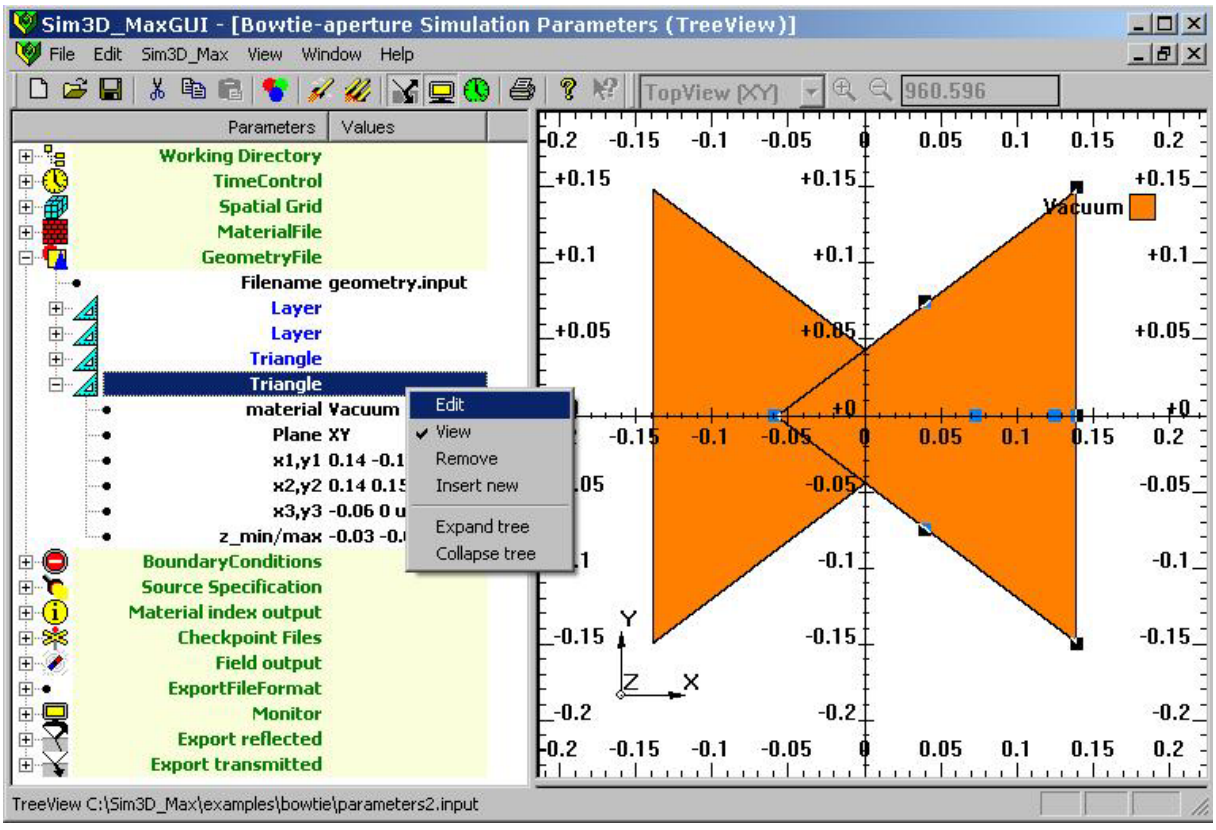
There are two ways to start an editing session for a displayed graphic shape:

- hold the CTRL key pressed and perform a left button click on the desired graphical shape
- within the TreeView, open the popup-menu (with a right button click) and select the ***Edit*** command, as indicated in the following picture.

When using the second way, notice the difference in the ***Edit*** command as compared with the other entries of the tree: instead of opening an edit-box in the TreeView (where the values can be typed in) the ***Edit*** command applied on a geometric primitive will attempt to enter the editing session in the GraphicView for that geometric primitive.

It should be noted that the edit-session for any shape is opened only if the geometric shape is physically displayed on the GraphicView (e.g. the ***View*** option must be enabled).

To terminate editing, simply click with the mouse on the background of the GraphicView or initiate editing for another geometric primitive object. Furthermore, only one geometric primitive may be edited at a time.



Sim3D_MaxGUI will reflect the start-up of an editing session by two measures: it will expand the TreeView entry of the geometric primitive under edit (collapsing the others) and it will redraw the GraphicView, placing some editing rectangle markers around and inside the edited shape, as displayed above, where the triangle to the right has been selected for editing.

Once this happens, any of the editing rectangle markers may be simply dragged with the mouse (place the mouse cursor inside the desired marker, press the left mouse button and keep it pressed while moving the mouse cursor). Depending on the dragged marker, one or more of the parameters displayed in the TreeView will change continuously during the drag operation. Simultaneously, the GraphicView will be updated, to reflect the changes.

Even if the geometry data applies to a three-dimensional environment, the changes will be done only in the two-dimensional plane actually selected in GraphicView (XY, XZ or YZ). This plane may be changed freely during an editing session, without the need to close and reopen the edit session for the geometry shape under consideration. Hence, the object can be viewed easily from any side during the editing. Furthermore, when the viewing plane is changed, depending on the specific geometry object, the editing markers will change too (that is, new ones may appear while others might be removed from the screen).

In addition to the self-explaining markers located on the corners or on the sides of shapes, for the most of the geometric primitives two new markers will be generated and displayed: one placed in the center of the shape, which serves to translate the object in the displayed plane, and one placed somewhat displaced from the center-marker (usually inside the shape, but occasionally outside), which can be used to rotate the shape around its center. These special markers will usually modify more then one parameter of the geometry primitive.

When two markers overlap exactly they will cancel each other out. However, the markers remain sensitive to the mouse cursor, even if not visible, and with some little practice it is still fairly simple to “guess” their location.

To keep the things simple, each marker has been attached to as few parameters as possible. Usually there is exactly one geometric parameter that gets modified when the corresponding marker is dragged, so the numerical changes can be followed easily. As a consequence, for example, around a rectangle shape at least two markers can be found: one for modifying the height and one for modifying the width. There may also be a third marker attached, which allows you to change both the width and height, during the same movement of the mouse cursor.

Once one of the markers has been captured by the mouse cursor (by pressing the left button down), it is kept captured until the mouse button is released, even if the mouse cursor leaves the markers rectangle. The numerical values and the display will change as long as the cursor remains inside the GraphicView area.

3.2.4. Material Color Pool-file

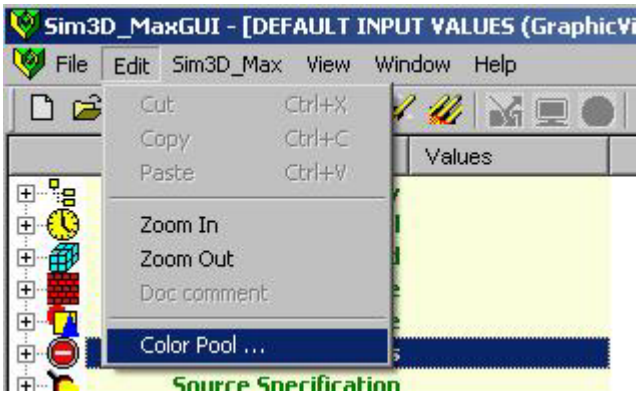
When Sim3D_MaxGUI displays the graphical primitive objects in the GraphicView, it assigns a color to each of them, based on its individual material name. It may happen that the colors of different materials differ only very slightly, almost imperceptibly to the eye. To improve on this situation, a simple material-color database has been implemented in Sim3D_MaxGUI, making it possible to assign arbitrary colors to specific material names.

The content of this internal database (map) is loaded at each startup from an external file, the Material Color Pool-file. The name and path of this file can be changed in the ColorPool-dialog box. Additionally, this dialog box offers the possibility to add/remove entries, to change the colors and/or to load/save other color pool-files.

The Color Pool dialog box can be invoked by one of the following commands:

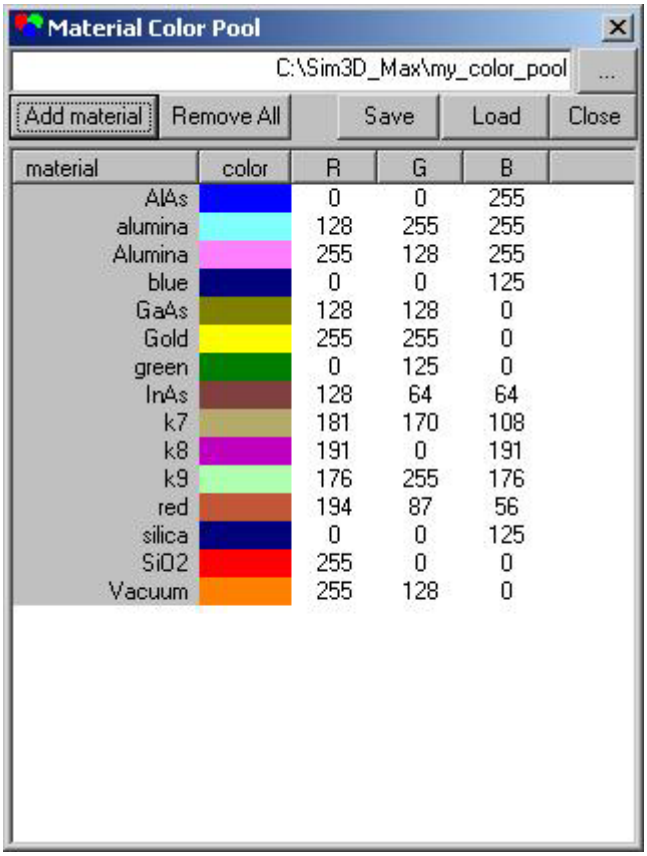
- by clicking the button on the toolbar.
- by selecting the main menu command ***File - Color Pool …***

- by selecting the main menu command ***Edit – Color Pool …***



- by selecting the popup menu command ***Color pool …*** within the GraphicView.

After invoking one of the above commands, the Material Color Pool dialog box should open, as illustrated in the picture below:



The following listing shows the content of the ***my_color_pool*** file, that was used to fill up the color list shown at the left:

```
k8,191,0,191
k9,176,255,176
silica,0,0,125
InAs,128,64,64
Alumina,255,128,255
Vacuum,255,128,0
GaAs,128,128,0
alumina,128,255,255
AlAs,0,0,255
SiO2,255,0,0
red,194,87,56
blue,0,0,125
green,0,125,0
Gold,255,255,0
k7,181,170,108
```

As it may be recognized, inside the dialog box, the material names are sorted in alphabetical order, which must not necessary be the case for the entries in the file.

Specifying a color pool file name

Loading of an existing color pool file can be initiated in two different but equivalent ways. Either type the file name directly into the edit box or click on the button labeled "**…**". If you click the "**…**" button, a FileOpen dialog box appears, allowing to select an existing filename. Note that in any case, no read/write operations are performed on the file, only the content of the edit box is changed to reflect the new filename.

Saving the color database into a file

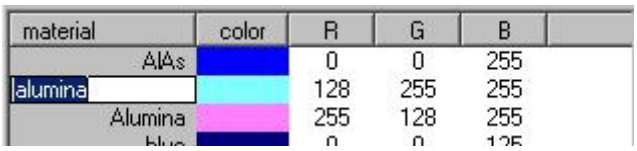
The file listed in the edit box, can be saved by clicking on the ***Save*** button. If the target color pool file already exists, Sim3D_MaxGUI will issue a prompt asking if the file should be overwritten.

Loading the color database from a file

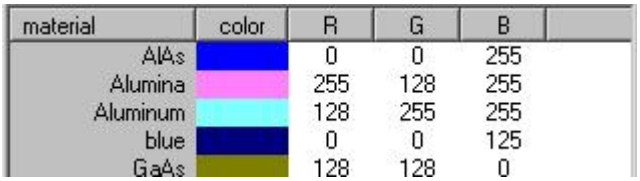
The file listed in the edit box, can be loaded by clicking on the ***Load*** button. If the source file does not exist, an error is issued; otherwise its content is merged into the existing pool. Note that the already existing entries are not removed before the new file is loaded. To clear all of the previous entries in the pool list use the ***Remove All*** button before loading a new file. If the new file and the existing list contain identical material names, then the color of that material is taken from the new file.

Changing the material name of an existing entry

To change the material name of an existing entry, simply click on the material name in the list and a small edit box will appear, as indicated below:



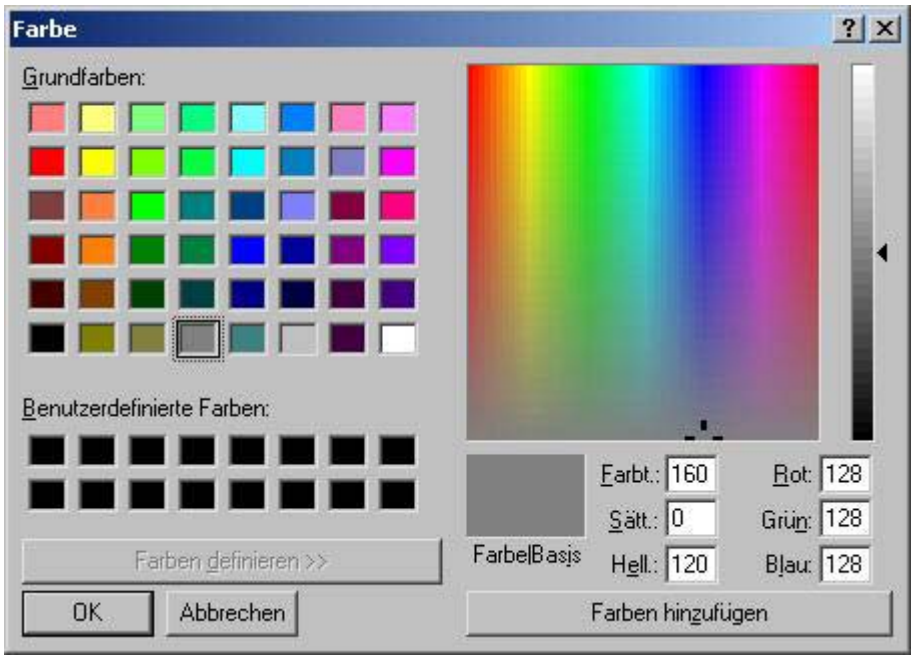
Type in the new, desired material name (for example “Aluminum”) and press ENTER or click somewhere else in the dialog box. As a result, the “alumina” entry will be replaced by “Aluminum”:



Note that the list is automatically re-sorted alphabetically.

Changing the material color for an existing entry

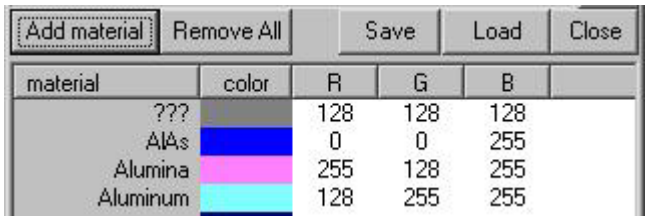
To change the material color for an existing entry, click on the colored rectangle to the right of the material name in the list in the column labeled “color”. A ChooseColor dialog box will appear, where the new color can be selected.



It should be noted that this dialog box is provided by the operating system that you are using. As a consequence, the labels are written in the specific language of the operating system version installed on your PC.
Optionally, you may modify the color by changing one of the R, G or B entries. To do this, click on the R, G or B column cell that you want to change, type in an integer value between 0 and 255 and press ENTER or click somewhere else on the screen.

Adding new material name entries to the list

To enter a new material name use the ***Add material*** button. When clicked, a new entry with material name "???" is added to the list. Since the list is sorting alphabetically the material names, this new entry will appear usually as the first entry, as shown below:



Before adding other new entries, you must rename the "???" to something else (let say "AluminumOxide"), and take into consideration the automatic re-sorting of the list. The ***Add material*** button will not add new entries until the "???" is changed to something else.

Removing entries from the list

To remove all entries, simply use the ***Remove All*** button. To remove one entry, rename the material name to an empty name and the line will be removed from the list.

Notes

1) At startup of a new Sim3D_MaxGUI session, the program attempts to reload the color pool file used in the previous session. If the file cannot be found or opened, the following error is issued:

When you encounter this error, ignore it at first, then open the color pool dialog box and specify an existing color pool file (it may be empty), by any of the methods described earlier in this chapter. Close the program and restart it, to verify that the error has disappeared.

2) Avoid using blanks inside the material names (for example, use "AluminumOxide" or "Aluminum_Oxide" instead "Aluminum Oxide")

3) Do not use "???" as a valid material name, since in this case you won't be able to add new materials to the list.

4) The material map is case-sensitive (for example “aluminum” and “Aluminum” are regarded as different materials)

5) During painting of the GraphicView, Sim3D_MaxGUI searches for each material name in the color pool first. If the material name is found there, that color is taken, otherwise the default algorithm is used to define the color.

6) The color pool file is not related to any other files used by Sim3D_Max or Sim3D_MaxGUI. This means that once a material name – color pair is specified in the color pool file, that color is used for all parameters.input and geometry.input files to display its associated material.

7) The material definition file is not connected to the color pool file. This means that you may specify the material names in each of the files independently.

8) Multiple color pool files can be merged together in the ColorPool dialog box by simply loading them one after the other and saving the result into the desired file.

9) Due to its simple syntax, the color pool file may be generated outside Sim3D_MaxGUI with any ordinary text-editing program.

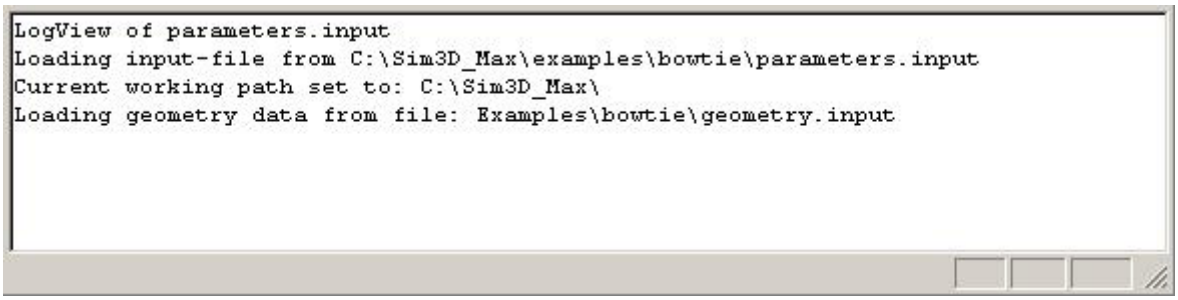
10) Any changes performed in the color list will remain valid at least for the actual session. To preserve them for subsequent sessions, save them back to the color pool file before closing Sim3D_Max_GUI.

11) Although relative pathnames may be used (like “.\mypool\filename”), it is recommended to specify a fully qualified name for the color pool file (like “C:\mypool\filename”) to avoid ambiguities.

12) Any extension (like “.txt” or “.pool”) or no extension at all (as in the above examples) may be used for the color pool filename.

3.3. The LogView

The only purpose of the LogView is to display the kind of information that is best packed into a text form. Errors, warnings or simply any additional hints for the user will show up inside this view.


LogView of parameters.input
Loading input-file from C:\Sim3D_Max\examples\bowtie\parameters.input
Current working path set to: C:\Sim3D_Max\
Loading geometry data from file: Examples\bowtie\geometry.input

It is recommended, at least when performing first steps with Sim3D_MaxGUI, to keep an eye on this view.

The LogView is very useful, when performing actions that require access to external files on the hard disk. In case of trouble, the LogView displays the attempted action and the reason why it failed. Furthermore, the LogView is the only place where the filenames are displayed with fully qualified pathnames, as shown in the example picture above. Please read the **Files and Directories** chapter of this manual to find out more about how Sim3D_MaxGUI deals with these items.

Additionally, the LogView gives more information about the content of the file being handled (like number of grid entries, size etc.).

4. The Data Views

During a simulation run, the Sim3D_Max program produces several output files, depending on the content of the parameters.input file supplied. To visualize the data contained in these output-files, Sim3D_MaxGUI provides numerous, additional specialized views, called Data Views.
Unlike the Main Views (TreeView, GraphicView and LogView), which are always displayed on the screen, a Data View must be explicitly invoked in some way, to be opened. Usually, this happens with the View-command of some Popup-Menu of the TreeView. Once opened, each Data View can be closed either like any other window, or by selecting again the View-command of the Popup-Menu.

One common thing to all Data Views is that they need some input data that is to be displayed. Usually, this data must be loaded first from the Sim3D_Max output files.

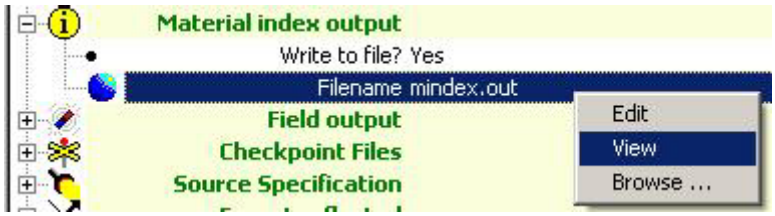
Some files may contain data that can be visualized in various ways (different cross-sections, for example). In this case the corresponding Data View offers the possibility to choose the desired drawing mode, and multiple Data Views can be opened to display the same data pool, but with different modes.

In the following subchapters, all available Data Views are documented. Since the Data Views have many common features, the first Data View will be described more detailed, while the other subchapter will focus mainly to the differences and special features of the respective Data View.

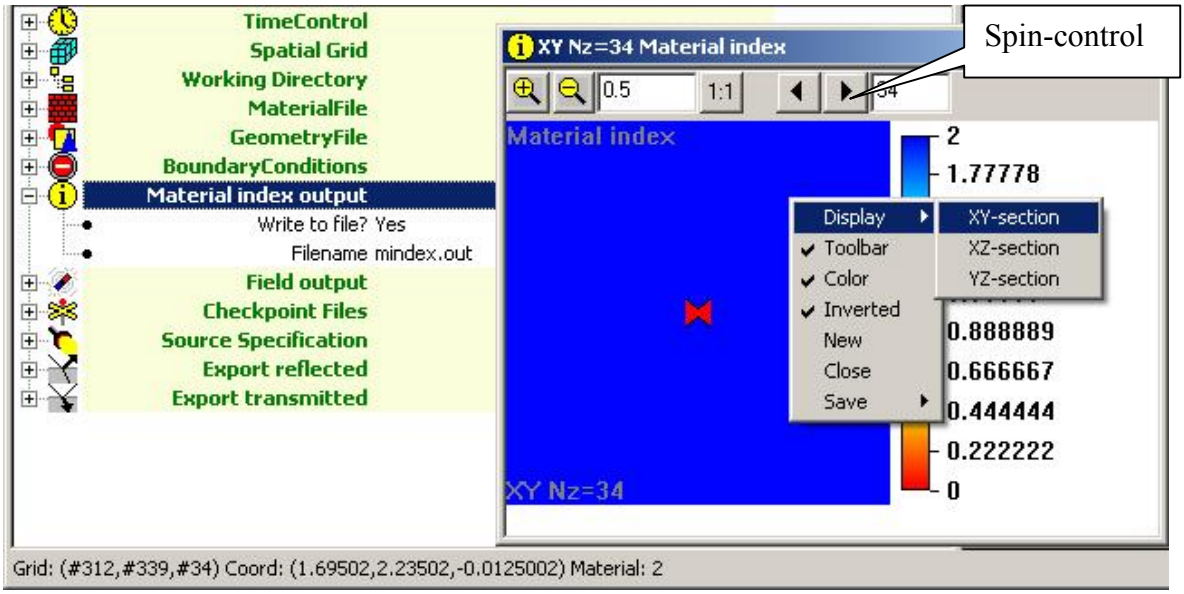
4.1. Material Index View

The material index files produced by Sim3D_Max contain integer numbers specifying the indices of the materials (not the refractive index!) for each point of the computational grid. With the Material Index View, those material index files may be brought to the screen.

The View is invoked by the View-commands of the Popup-menus of the "***Material index output***" or the "***Filename***" entries of the TreeView as shown below.



After selecting one of the above commands, Sim3D_MaxGUI loads the material data from the specified file and opens the View.



The Material Index View displays cross-sections of the tri-dimensional grid as bitmaps. From the pop-up menu (invoked with a right-click) one may select the desired section (XY, XZ or YZ), as shown in the picture. When the mouse moves over the bitmap, some information

about the location inside the computational grid and the value at the cursor position is displayed in the status bar of the main window. Sim3D_MaxGUI takes the coordinate values for each grid point from the *xcoord*, *ycoord* and *zcoord* files produced by Sim3D_Max.

The index of the free coordinate (in the above picture this is Nz) may be selected with the spin-control of the View-toolbar.
The other controls of the toolbar are used to alter the size of the bitmap on the screen: the 1:1-button reserves for each grid point one pixel (this corresponds to a zoom factor of 1) and the + and – buttons double or halve the size of the bitmap. It is recommended that you use the 1:1-button immediately after first opening of the View , in order to establish a 'natural' size of the bitmap.
If you want to set an arbitrary zoom factor (greater then 0) then you have to type the desired value into the zoom-edit-box and press ENTER to cause Sim3D_MaxGUI to take action.
The Pop-up menu offers the following commands:

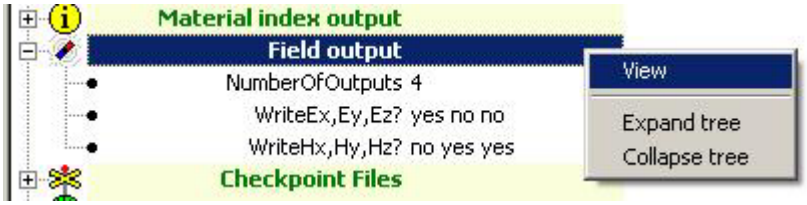
- **Toolbar** – displays or hides the toolbar of the window.
- **Color** - switches from colored bitmap to a grey-scaled bitmap and vice versa.
- **Inverted** – inverts the colors (or grey levels) of the bitmap, creating a 'negative' view.
- **New** – Opens a second Data View that will display the same material index file. This is useful, if you want to see all possible cross-sections (XY, XZ and YZ) simultaneously on the screen.
- **Close** – Closes the Data View. If there is more than one View opened, only the one chosen View is closed.
- **Save** – This menu entry has two subcommands, data and bitmap, which allow you to store the actual displayed cross-section as a data-matrix or as a BMP-file.

Notice that these menu commands are common to all Data Views that display bitmaps.

There are two ways to close the Data Views: by using the Close-command (the Windows-way) or by again using the View-command from the Popup-menu of the TreeView (the same way as used for opening). It is important to know that there is a small, but sometimes valuable, difference between them, concerning mainly the handling of the memory reserved to keep the data from the visualized file.
When closing the Data View in the usual way (with the Close-command), then the memory is freed and the Data View is destroyed. As a consequence, when reopening the Data View the data must be reloaded into memory from the disk-file, and one has to wait. Furthermore, the size and the settings of the Data View will be lost, and it will be displayed again with default size and settings.
When closing with the View-command, the memory is not released and when the Data View is invoked again, it will show up quickly. In addition, the size and settings are preserved, making its usage more comfortable, at cost of memory.

4.2. Field Output View

The Field Output View works similar to the Material Index View, but with some differences, explained in the following.
It always expects a subset of the following files to be present: *Ex.bin.out*, *Ey.bin.out*, *Ez.bin.out*, *Hx.bin.out*, *Hy.bin.out* and *Hz.bin.out*. These names are fixed, and cannot be changed. To open the Field Output View, select the ***View*** command from the popup menu of the ***Field output*** entry of the TreeView, as displayed below:

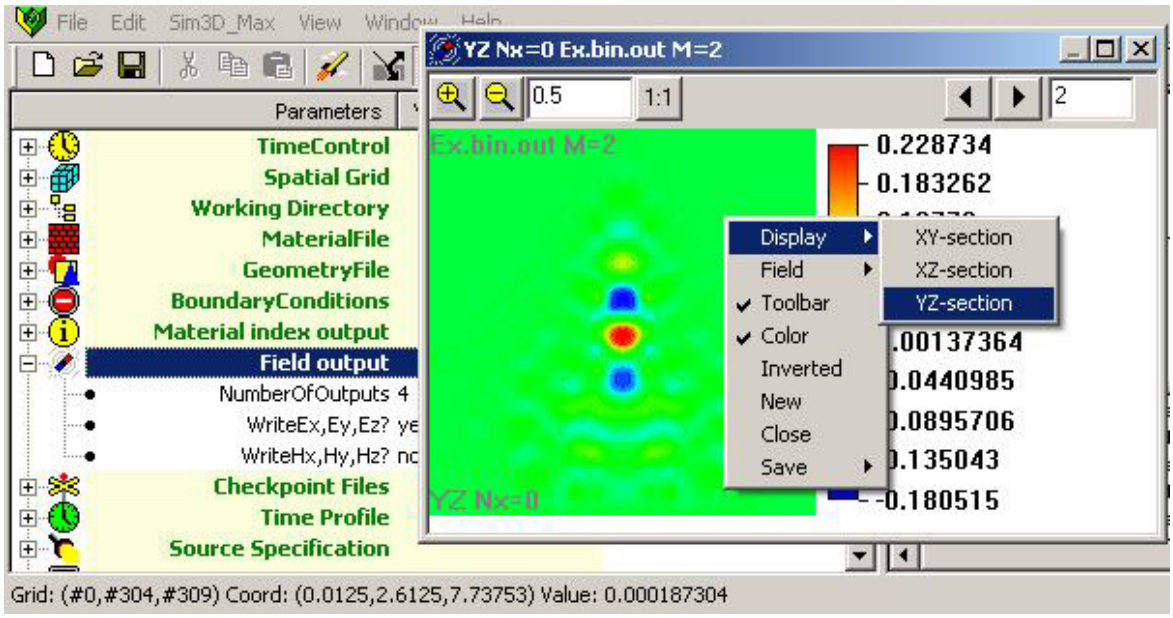


Sim3D_Max places in the field output files snapshots of the E and H field values over the whole computational domain, taken at different times. The structure of the files is very similar to the material index files, with the main difference being that the field output files may contain many computational domains, arranged sequentially, like a small movie. The number of frames (or snapshots) is given by the ***NumberOfOutputs*** parameter.
Hence, the Data View for these files will offer the same functionality as the Material Index View, but with the following additional features:

- The toolbar contains one additional spin control, which allows you to select the 'frame of the movie', in other words the number of output (noted **M=2** in the example picture below)

- The popup menu has an additional command Field that allows you to select one of the Ex, Ey, Ez, Hx, Hy and Hz fields to be displayed. Note that all of the menu entries are always present even though not all field-output files are available. Those not available will be grayed out and will not be selectable.

The following picture shows a Field Output View displaying the third frame (M=2) of an *Ex.bin.out* file.

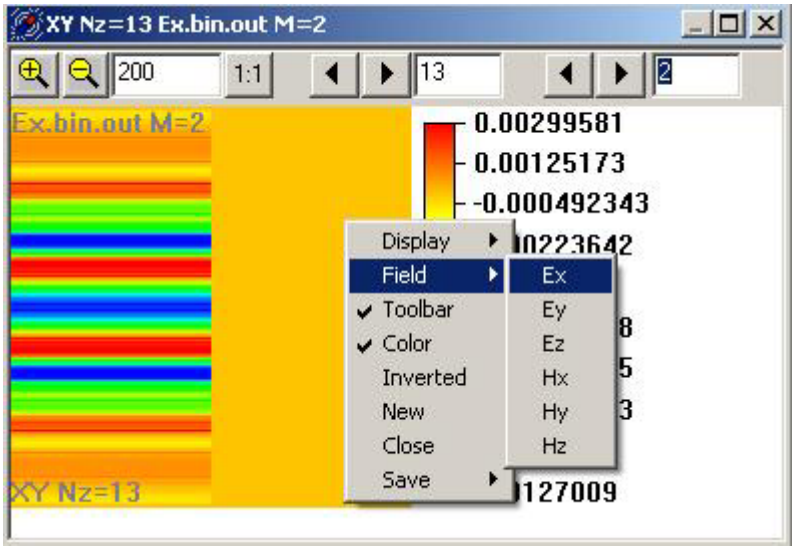


Note that in the above example, the computational grid has only one layer in the x-direction, therefore the spin control for Nx is hidden and does not appear in the toolbar. (Nx is always 0 here). Furthermore, the displayed Ex.bin.out contains 4 frames (outputs), that means M can take the values 0,1,2 and 3 (change this values with the rightmost spin control that is displayed in the toolbar).

The next picture displays the XY cross-section of the same file. Two things are important to be noticed:

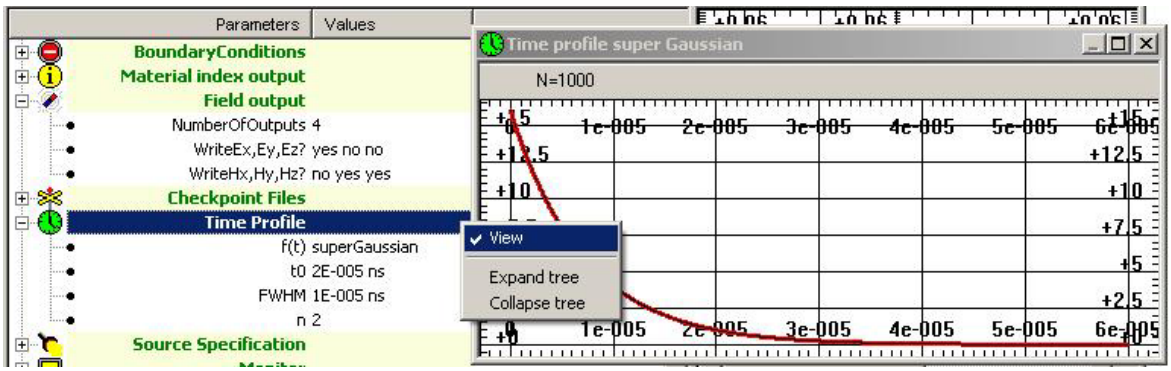
First, the computational grid has more than one layer in the z-direction, thus the spin control for selecting the Nz parameter is enabled and displayed on the toolbar.

Second, as mentioned before, the computational grid has in x-direction only one layer. As a consequence, the bitmap displayed on the screen would be only 1 pixel wide and hence difficult to usefully display in both X and Y. In this special case, the bitmap is kept at a size greater then 50 pixels, so each pixel of the de facto 1 pixel wide bitmap appears in fact as a 50-pixel-wide line.



Finally, it should be noted that Sim3D_MaxGUI needs to decipher the field output data stored by Sim3D_Max in the *xcoord*, *ycoord* and *zcoord* files. If these files happen to be corrupted, Sim3D_MaxGUI is not able to display the field output correctly. When this happens an error message will be issued in the LogView.

4.3. Time Profile View



The Time Profile View differs somewhat from the other Data Views because it does not need any output files from Sim3D_Max. It has been implemented, as a convenience, to show the shape of the time profile, as it is input into the ***TimeProfile*** entry of the TreeView. As such, it displays the f(t) modulation function that is selected (in the above example this is the superGaussian function, taking the values from the TreeView. The function formulas used to compute the curve are the same as those given in the Sim3D_Max manual. In the case of a user defined time profile the data is taken from the specified *user_time_profile.input* file (if available).

The view can be opened, like the other Data Views, by invoking the ***View*** command of the popup menu of the ***Time Profile*** entry in the TreeView.

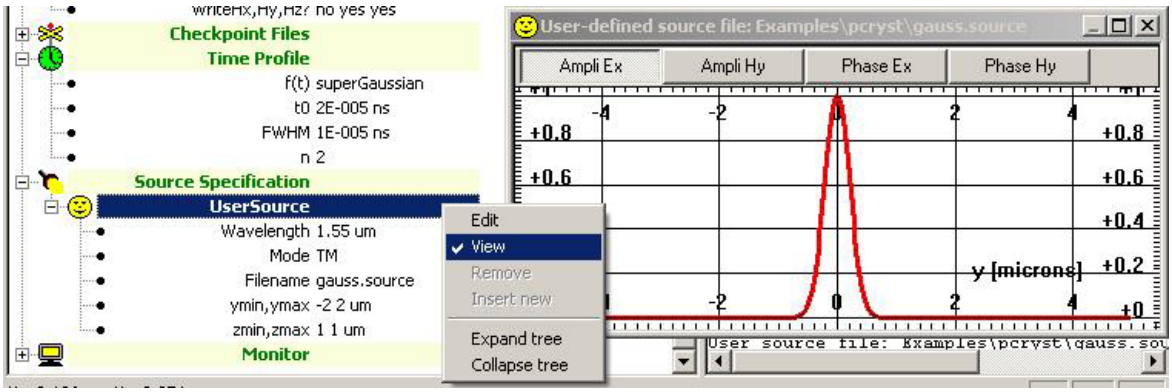
Once opened, it will reflect immediately any changes of the parameter values of the TimeProfile entry, providing a quick and comfortable tool to verify if the shape meets the expectations.

4.4. Source Views

Sim3D_Max accepts several different source specifications as input. Two of them may be visualized with specialized Data Views: the User Source and the DIFFRACT source.

4.4.1. User Source

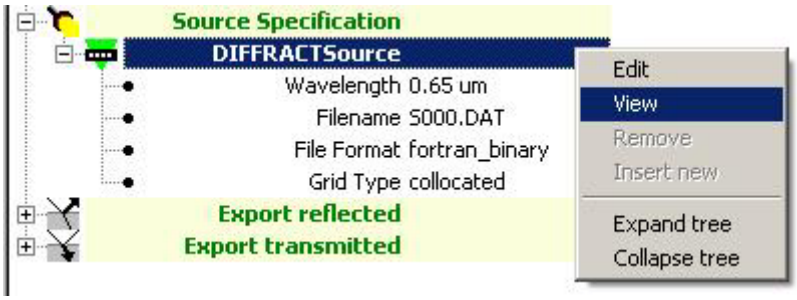
For two-dimensional computations Sim3D_Max accepts arbitrary source distributions that can be read from *usersource.dat* files. The content of such files can be visualized with the UserSource View. It can be opened with the View command of the popup menu of the UserSource entry of the TreeView. An example is shown below:

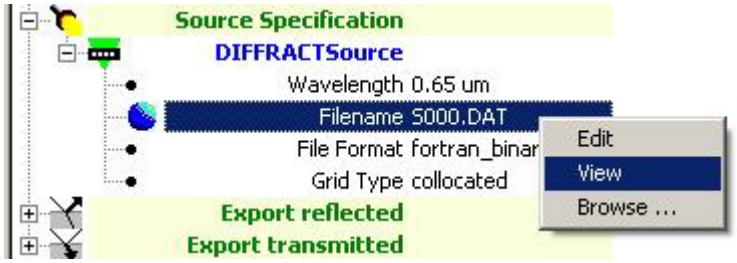


The UserSource View plots the amplitude and/or phase of one or both of the fields read from the *usersource.dat* file against the coordinates read from the same file. Note that the fields are expected to be stored in the *usersource.dat* file as complex numbers (real and imaginary values for each field), but Sim3D_MaxGUI converts the real-imaginary components into amplitude-phase pairs.
Select the curves to be displayed from the toolbar buttons. Every curve is displayed with another colour, so they can be discerned easily on the screen.
The name of the fields and the abscissa coordinate label change accordingly to the ***Mode***, ***ymin,ymax*** and ***zmin,zmax*** values specified in the UserSource entry of the TreeView. Please refer to the Sim3D_Max manual for details.

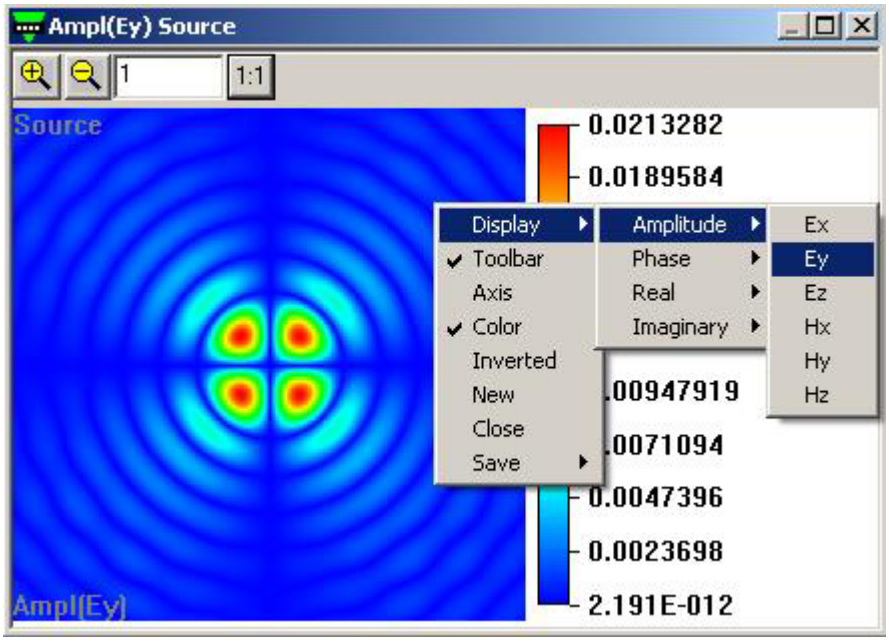
4.4.2. DIFFRACT Source

Within this DataView, the source files created by DIFFRACT™ can be visualized as bitmaps. Open the DIFFRACT Source View in the usual way, by using the ***View*** command of the popup menus of the ***DIFFRACTSource*** or ***Filename*** entry of the TreeView, as displayed below:





The data format is described in the DIFFRACT manual, please consult that document for further information.

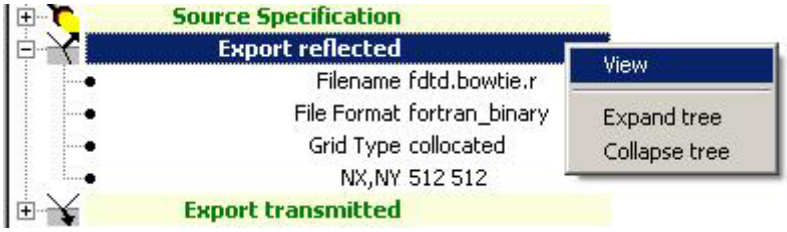


Once the View is opened, with the popup menu any one of the components can be selected to be displayed (see above example picture). In addition, the popup menu has the command ***Axis***, permitting the DIFFRACT Source View to draw a coordinate system into the bitmap. It is important to note that the coordinates will be normalized by the wavelength that is specified in the TreeView.

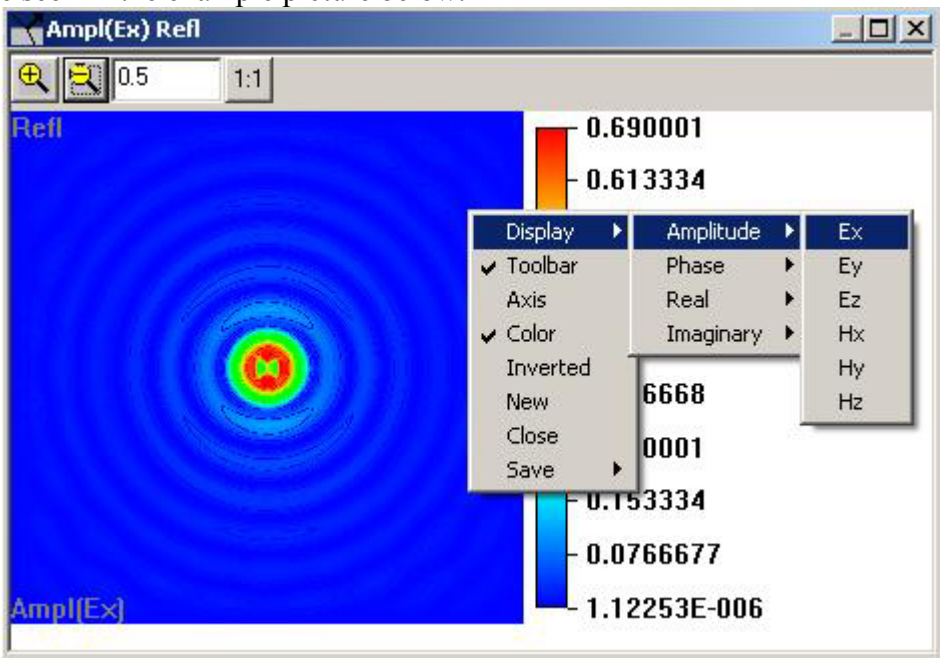
Furthermore, the ***File Format*** parameter is taken into consideration: if the format of the file does not match the format specified in the TreeView, then Sim3D_MaxGUI is not able to read the source file.

4.5. Export Reflected View

This view allows to visualize the files created by Sim3D_Max with the ExportReflected command. Open the view as exemplified below:

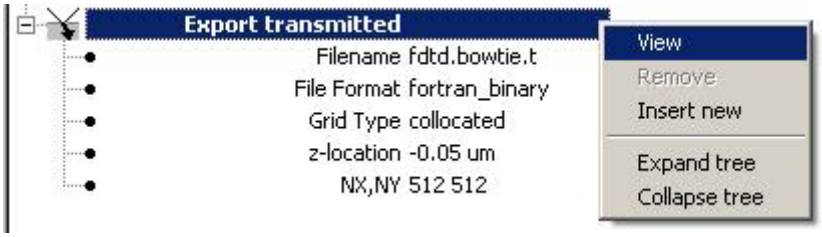


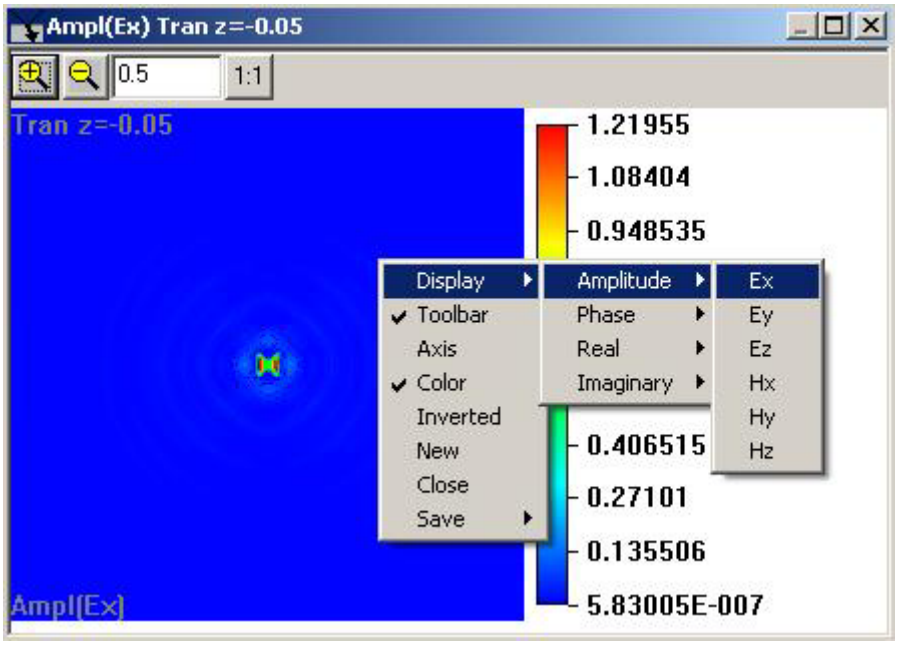
The data format is identical to the format of the DIFFRACT source files, therefore this view is identical to the DIFFRACT Source View and everything said about viewing DIFFRACT sources remains valid.
This can be seen in the example picture below.



4.6. Export Transmitted View

The Export Transmitted View is practically the same as the Export Reflected View and the DIFFRACT Source View. For completeness, the following pictures show how to open the view and an example-screenshot.





4.7. Monitor View

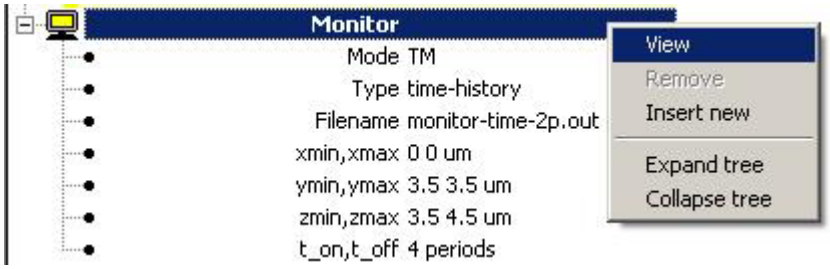
As the name suggests, this view displays the content of the files created by Sim3D_Max during execution of the Monitor command blocks. The data format depends on the Mode and Type options, specified in the Monitor command block. These options are stored by Sim3D_Max in the monitor files and the Monitor View take them from the files and not from the TreeView entries.

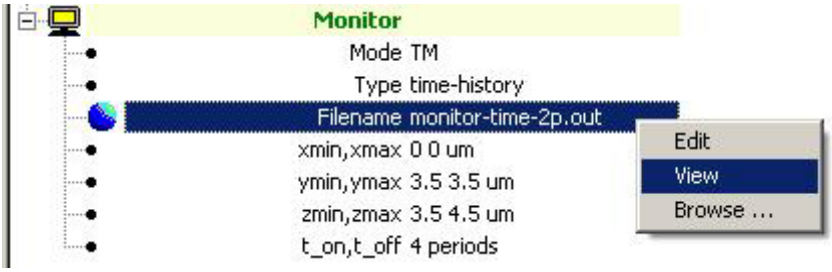
In contrast to the other DataViews, for each monitor block there can be more than one file present. This is the case when the simulation run has been done on a multiprocessor machine in parallel mode: each processor can create his own output file for the same Monitor command block. To discern the files, each processor appends its index to the filename.

Consequently, before opening a MonitorView, Sim3D_MaxGUI searches the output directory for output files of up to 10 processors. It loads the founded files and merges their data into one single, homogeneous, internal block, so there is no need to manually keep track of how many processors were involved in the computation. However, it should be mentioned here, that the file merging can only be done if all files have been created during the same simulation run.

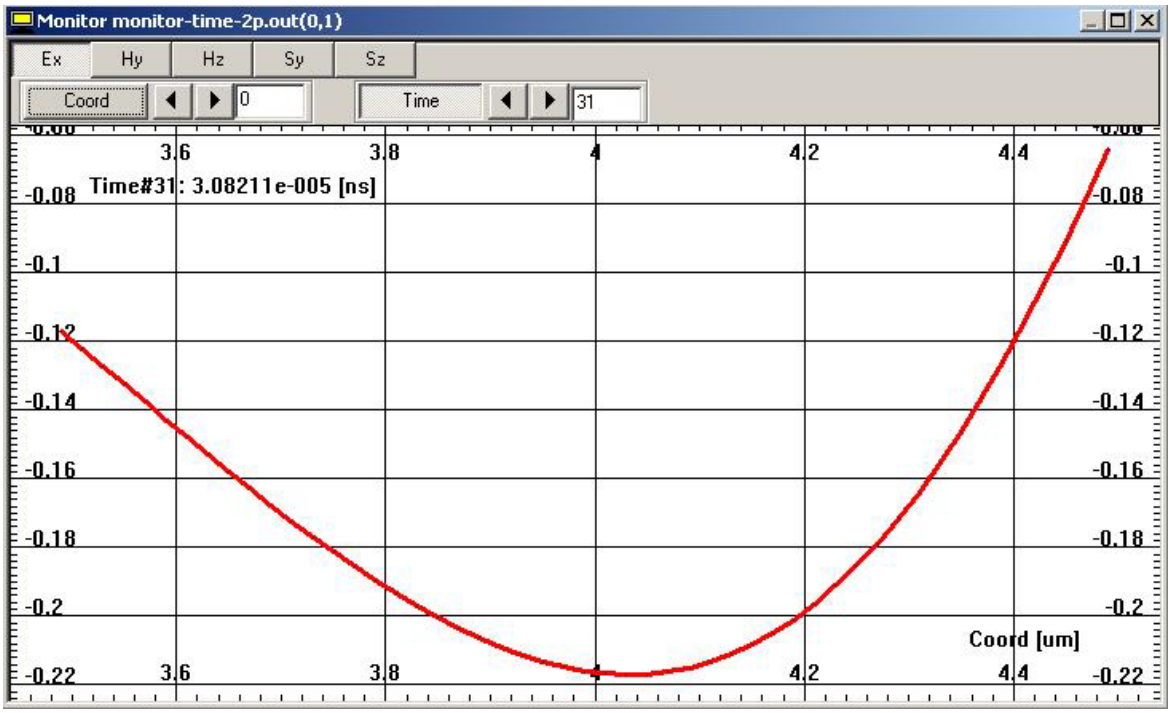
The monitor files contain field values sampled at specified points and within a specified time or frequency interval. The Monitor View displays the field samples as curves: it plots the values against the coordinate values or against the time/frequency values.

Open the Monitor View as usual, with the ***View*** command of the popup menu attached to the ***Monitor*** or ***Filename*** entries of the TreeView (see pictures below).





The next picture shows the Monitor View opened with one of the above commands. In this case two processors have been involved in the computation: processor 0 and processor 1. This is indicated with the (0,1) bracket in the caption text of the window.



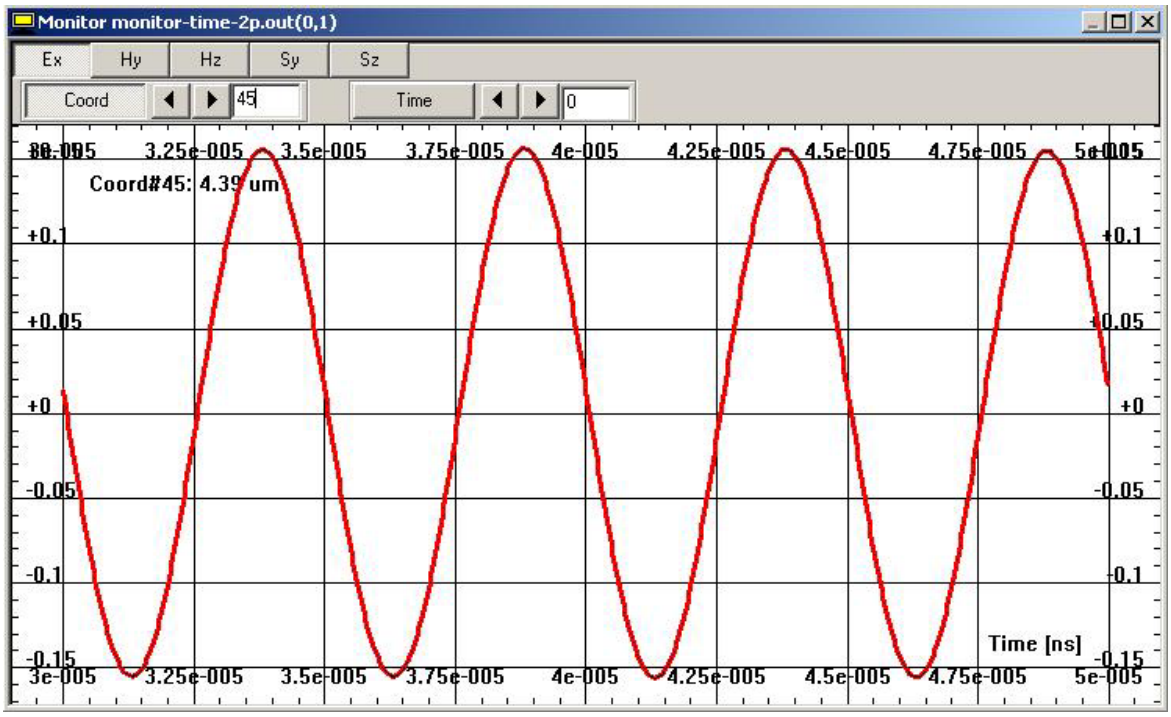
The first series of buttons inside the toolbar (the Ex, Hy, Hz, Sy and Sz buttons in the above picture) allow you to select the field value to be displayed. Note that the number and the labels of those buttons vary, depending on the Mode and Type of the loaded monitor file. For each selected field type a curve is generated from the bulk of monitor data merged from the monitor files.

Since Sim3D_Max stores the field values for each point in space and time (or space and frequency, depending on the Mode and Type), there are two possibilities to plot the field values:

1. Against the space coordinates, in which case one could sweep deliberately through the time points (e.g. for each time there is a different curve plotted against the space coordinates)

2. Against the time (or frequency), in which case one could select freely one of the available space coordinates (e.g. for each space point there is an independent curve plotted against the time or frequency interval)

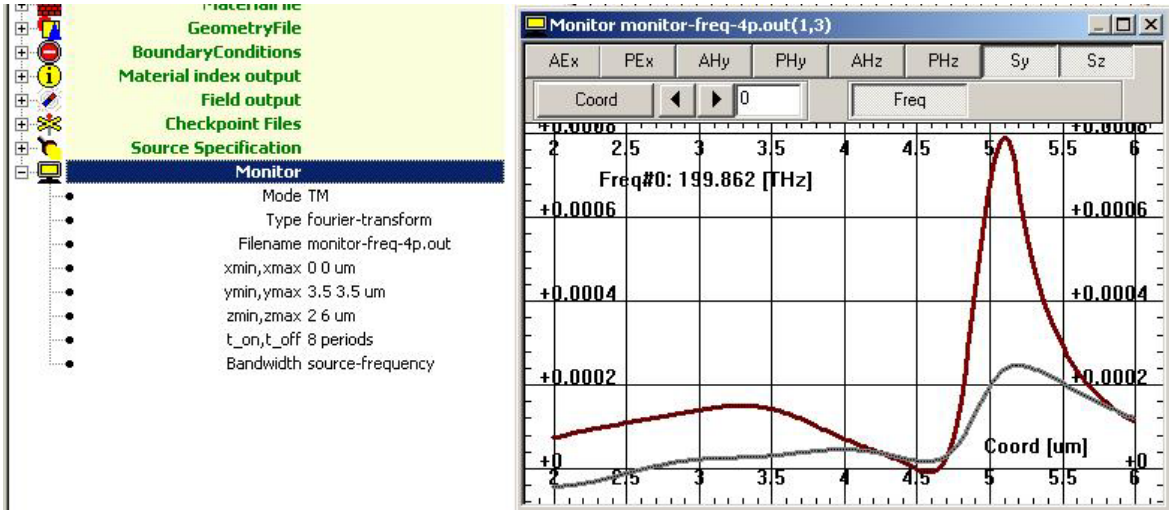
The previous picture displays the first plot mode (field values plotted against spatial coordinates), and the picture below shows the same data, but this time in the second plot mode (field values plotted against time).



The following should be noted:

- to plot against spatial coordinates, click on the ***Time*** button and use the spin control attached to the ***Time*** button to select one of the time points. In this mode, the spin control attached to the ***Coord*** button is meaningless.
- to plot against time, click on the ***Coord*** button and use the spin control attached to the ***Coord*** button to select one of the coordinate points. In this mode, the spin control attached to the ***Time*** button is meaningless.

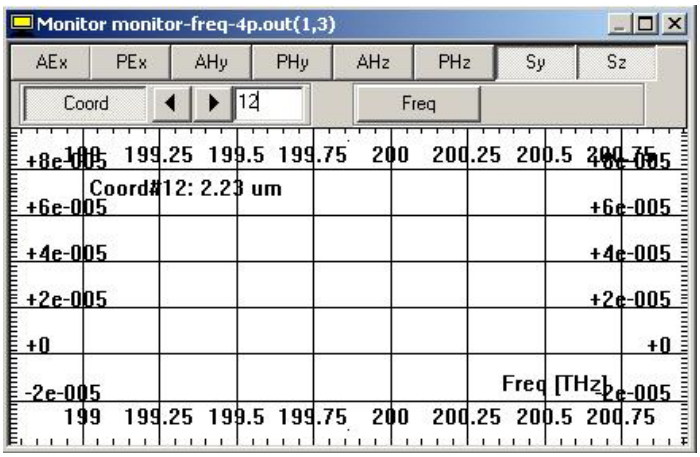
As another example, look at the picture below.



Since the Type is fourier_transform, the ***Time*** button is replaced by the ***Freq*** button, but the functionality remains the same. Furthermore, the data has been recorded at only one frequency point (the frequency value is displayed in the **Freq#0: 199.862 [THz]** label). This

has two consequences: first the spin control near to the Freq button disappears and, second, the only meaningful plot mode is against the coordinates (as displayed in the picture).

Note that ***Coord*** button and its spin control are still visible and useable because, even if in this mode (plot against the one and only frequency point) no plot would be drawn, the numerical values of the coordinates are displayed in the graph area. Hence, one could read out each coordinate value by sweeping with the spin control (see picture).

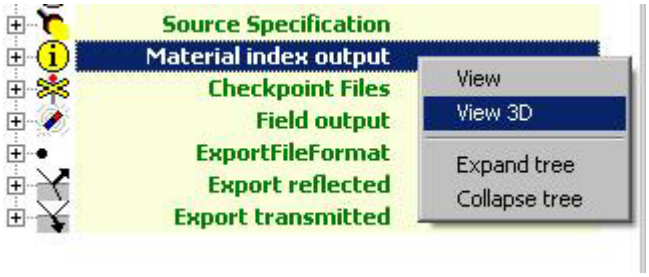


In this example, some field values are stored as complex numbers. Sim3D_MaxGUI converts this numbers into amplitude-phase pairs and assigns the letters A or P, standing for Amplitude or Phase. Hence, **AEx** means 'Amplitude of Ex', **PEx** means 'Phase of Ex', etc.

5. The Data 3D Views

Since Sim3D_Max performs its simulations in a tri-dimensional computational grid in space, many of the generated output data files rely on this grid, containing specific information for each of its points. As mentioned in the previous chapter, this data can be visualized in the specific Data Views for each type of data, where sections of the computational domain are displayed as two-dimensional plots. The Data 3D Views eliminate this restriction by drawing the data in a 3D scene on the screen.

Most of the input/output files that can be displayed in a Data View have the additional option to be represented in a 3D View. To use this facility, select the ***View 3D*** command in the corresponding popup menu of the TreeView, as illustrated below.



There are two kind of Data 3D Views: *solid scene* and *relief scene*, which will be described more detailed later in this section. In the solid scene view, the data is used to build up tri-dimensional objects which are then displayed within the computational domain box, hence this type will be used to display the material index file and the various exported fields. In contrast, the relief scene will accept as input a two-dimensional data field which is then displayed as a mesh, having the nodes elevated. As such, it is well suitable for displaying two-dimensional beam distributions and fields.

Once the ***View 3D*** command is invoked, the view loads the corresponding data and opens on the screen. Note that the loading process may take some time (up to few minutes, depending on the file size and type).

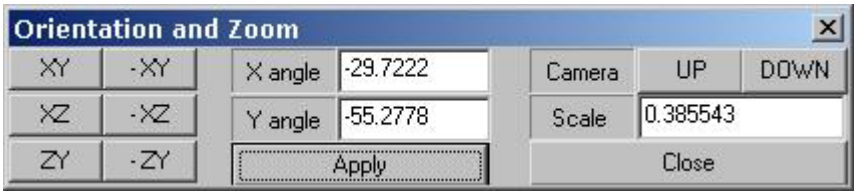
After the view is opened, a popup menu may be invoked with a mouse right-button click inside the View3D window area. This menu will contain all necessary commands to control the aspect of the view. Generally, each of menus commands will open some helper dialog box. First, the features that are common to both types of View3D (*solid scene* and *relief scene*) will be described, followed by a separate description of the specific features for a *solid scene* and *relief scene.*

5.1. Common Features

All View3D types offer the possibility to change the position and orientation of the scene, regardless of its type. Furthermore, the colorizing of the data follows some predefined color legend, which can also be specified, in certain limits, for all kind of View3D scenes.

5.1.1. Orientation and Zoom

This dialog box can be invoked by specifying the ***Orientation and Zoom*** command in the popup menu.



Within it, the orientation of the displayed coordinate axes of the scene can be rotated with respect to the (imaginary) coordinate axes that are defined by the screen surface (x-axis pointing to the right, y-axis pointing up and z-axis pointing out from the screen to the viewer).

The corresponding angles, labeled *X angle* and *Y angle,* can be modified arbitrarily. After changing them, the *Apply* button must be clicked, or the *Enter* key on the keyboard pressed, for the changes to take effect.

More simple, the buttons labeled with *XY, XZ, ZY, -XY, -XZ* or *–ZY* will position the viewing point (camera) at predefined position on the axes. For example, the *XY* button will place the camera on top of the z-axis, while the *–XY* button will place the camera in the negative end of the z-axis.

The viewing point can be moved closer to the scene with the *UP* button, which results in a bigger scene. The reverse effect can be achieved with the *DOWN* button. More directly, the distance of the camera can be specified directly by the *Scale* factor and a click on *Apply* or a hit on *Enter*.

However, the usual and more practical way to change the orientation and size of the scene is with the mouse.

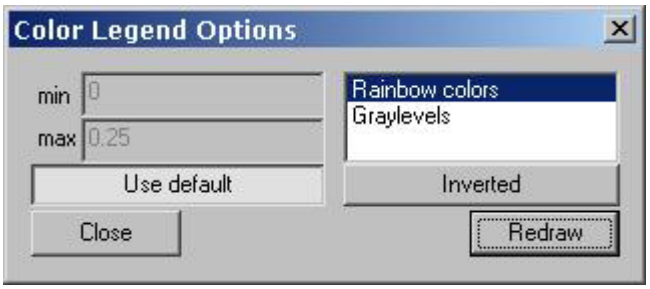
To modify the orientation simply drag with the left mouse button on the screen surface (press the left button of the mouse, hold it pressed and move the mouse). All rotations occur always around the center of the screen, where usually the center of coordinate axes lies.

To translate the scene, hold the CTRL key pressed down and drag with the mouse over the screen, as described above. It should be noted that the mouse cursor movement on the screen is always coupled to a x- and y-translation along the x and y coordinate axes of the scene.

To change the size of the scene use the scroll wheel of the mouse or the arrow-up and arrow-down keys on the keyboard. This is similar to the UP and DOWN buttons of the dialog box.

5.1.2. Color Legend

This feature allows to specify the colorizing of the displayed data. The dialog box can be opened by invoking the ***Legend…*** command on right-button popup menu of the 3D view.



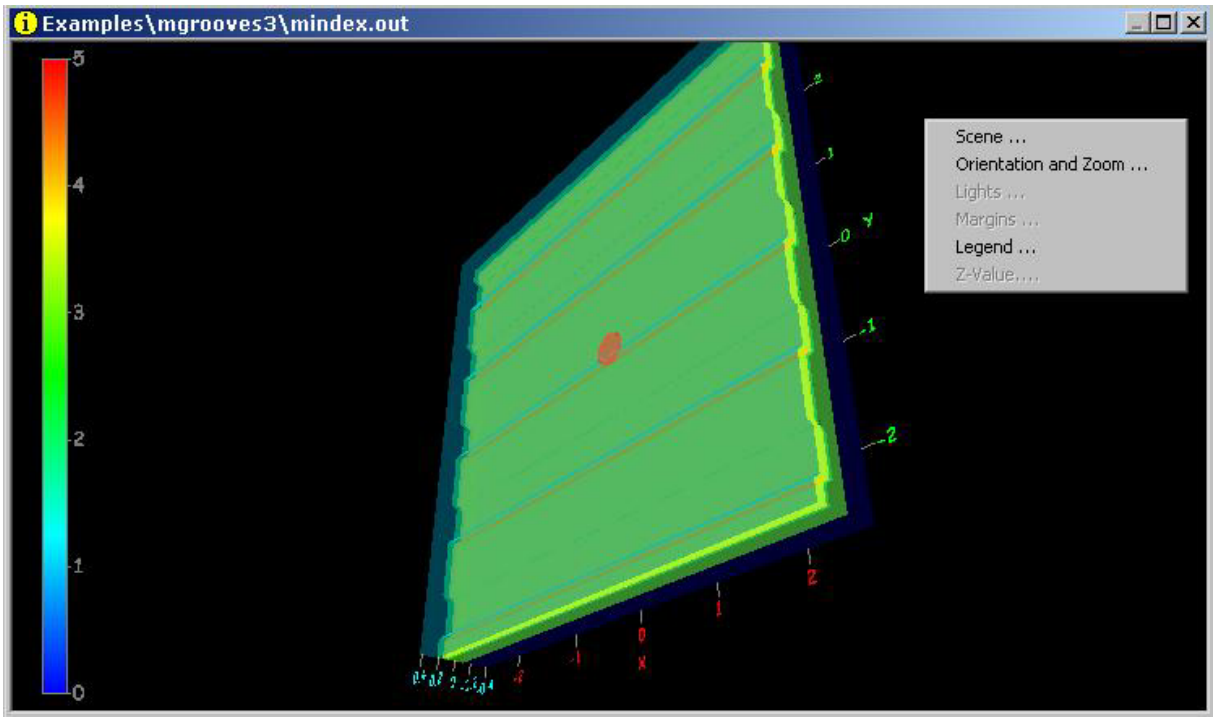
When the option *Rainbow colors* is selected, the lowest values will be displayed blue and the highest red. The *Graylevels* option will display the lowest values black and the highest white.

The option *Inverted* will revert this functionality in both cases, that is, for *Rainbow colors* the lowest values will be displayed red and the highest blue. Analogously, for *Graylevels* the lowest values will be white and the highest black, in this case.

By default, the lowest and highest values are determined automatically from the displayed data. This will happen if the *Use default* option is checked. When this option is unchecked, however, then the min/max values can be specified arbitrarily and the colors or gray levels are spanned over these values. Data values less then the min value will then be displayed all in the color used for the min value (blue for non-inverted Rainbow colors, black for non-inverted gray levels) and data values greater then the specified max value will be displayed all in the same color as used for the max value (red for non-inverted Rainbow colors, white for non-inverted gray levels).

5.2. Solid Scene

The *Solid Scene* view 3D type is used to display volumetric data over the computational domain. Material index files or exported fields serve typically as data input for this kind of scene. As such, it is selected automatically when the ***View3D*** command of the popup-menu attached to the *Material Index Output* or *Field Output* entry of the TreeView is invoked. The following picture illustrates this scene type for a material index output file generated by Sim3D_Max for a grooved layer geometry sample.



In addition to the data file, the *Solid Scene* attempts always to load the coordx, coordy, coordz and coords files generated by Sim3D_Max, which are necessary to display correctly the computational domain box.

5.2.1. Loading and preparing the data

Generally, the scene is build up by rectangular boxes placed inside the computational domain box. Unfortunately, the trivial approach to use one small box for each computational point is not applicable, as some simple math shows: for example, a reasonable sized grid of 300x300x300 has 27 millions of grid points. Assuming roughly 1 ms for rendering and displaying the box for one single point will result in prohibitive 27 thousands of seconds necessary to display the whole grid.

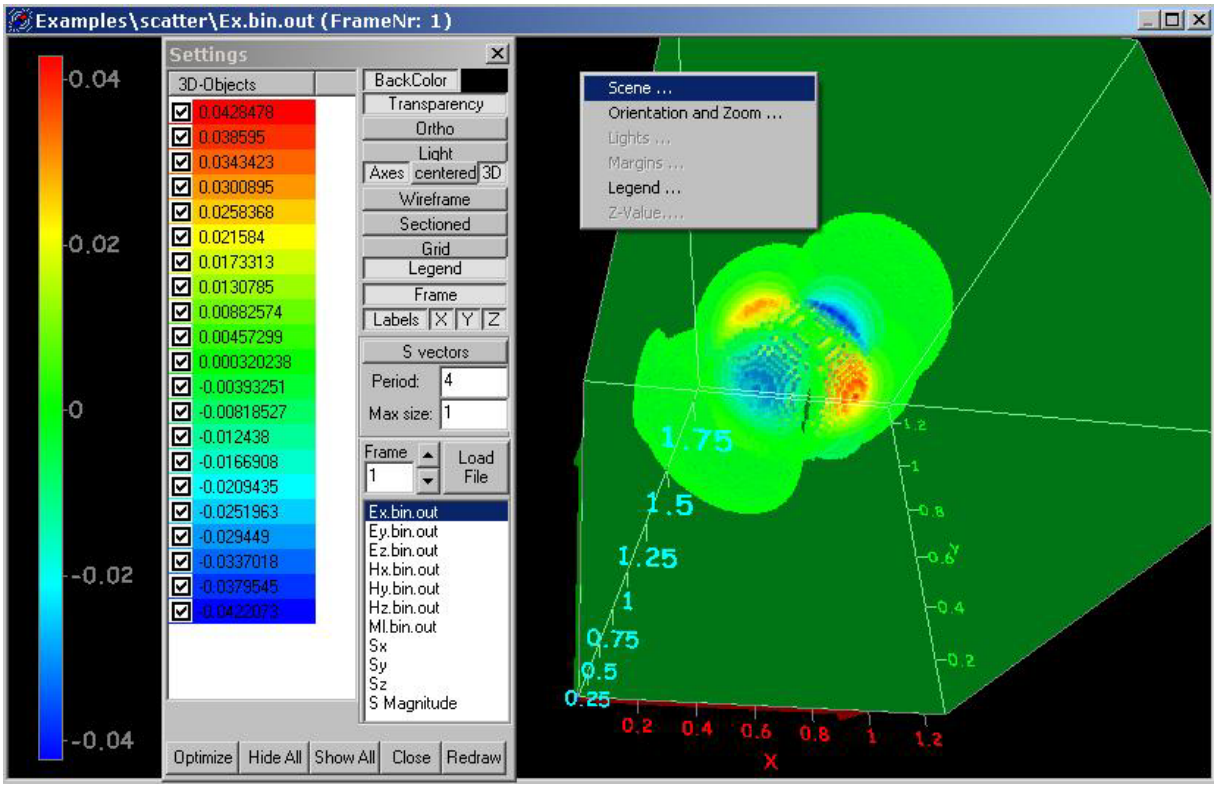
To face this problem, all neighbored boxes that represent the same data values (bearing therefore identical colors) are joined together building up larger boxes. Since in a typical data file, there are only few different values, this will reduce considerably the number of boxes that are displayed in the scene, without any loss of information. However, this must be paid with some extra time at the loading of the data, time spend to find the neighbored identical grid-point-boxes and bind them together properly.

The boxes that represent the same data value but cannot be glued together (because they are disjoint) are grouped into one cluster, called 3D-Object. The whole solid scene is build up by

one or more such 3D-Objects, each of them being associated with a different data value. In other words, there will be as many 3D-Objects as many different data values are found in the file. When the data file is a material index file, the number of different values equals the number of different materials used in the geometry. However, in the case of an exported output field file, since many different data values occur, an unacceptable large number of 3D-Objects would result. To avoid this, in this case the number of 3D-Objects is limited to 32 and a certain 3D-Object is build up by all boxes having values that lie within a limited range (e.g. their values must not be identical).

5.2.2. Scene Options

The ***Scene Options*** dialog box may be invoked with the ***Scene…*** command of the popup menu, as illustrated below.



Due to its many control elements, this dialog box is one of the most important and should be used extensively. There are three kind of controls displayed: the 3D-Objects list, controls that modify the scene itself and controls that specify the displayed data, which will be described in the next, followed by some specific example pictures.

- **BackColor** – when this option is specified, then the color displayed in the associated rectangle will be used for the background of the scene. To change this color, click on the associated colored rectangle and select a new color from the standard ChooseColor dialog box that opens. When this option is not specified, the default color given by the Windows operating system for the windows background is used.

- **Transparency** – will determine the framework to draw all the 3D-Objects transparent.

- **Ortho** – selects an orthographic projection. In contrast, when this option is not selected, a perspective projection is used. These two projection modes differ only by the way they consider the depth information: the perspective projection consider the depth (which means closer objects are displayed bigger and vice versa, far objects will be displayed smaller), while the orthographic projection ignore the depth (which means the displayed size does not depend how near or far an object is)

- **Light** – enables or disables (e.g. turns on or off) 6 light sources used internally to illuminate the scene. Usually, for best results, when the light is on then the transparency should be disabled (that is, the objects are displayed opaque). For more information, read later in this section.

- **Axes** – enables or disables the drawing of the coordinate axes in the scene. It should be noted that the x-axis is always displayed red, the y-axis green and the z-axis blue.

- **Centered** – forces the framework to place the origin of the coordinate axes always in the middle of the screen.

- **3D** – changes the aspect of the axes from wire-frame arrows to solid arrows.

- **Wireframe** – will force the framework to display the 3D-Objects as wire-frames. It should be noted that in this case *transparency* and *light* make not much sense, so they should be disabled when wire-framing is selected. Furthermore, this button will force ALL 3D-Objects to be displayed as wire-frames. However, it is possible to select only a subset of the 3D_Objects for being displayed as wire-frames. Please read the description of the 3D-Objects list for more details.

- **Sectioned** - when this option is selected, three, perpendicular to each other, cross-sections of the scene will be displayed instead of the 3D-Objects. The cross sections are build up from the actual scene data. Each section plane can be moved along the axis perpendicular to it by using the X, Y, Z, Shift+X, Shift+Y or Shift+Z keys on the keyboard. The actual values will be displayed as Xs, Ys and Zs on the screen. These values cannot be chosen arbitrarily, they follow the values taken from the coordx, coordy and coordz files (generated by Sim3D_Max™) found in the project directory. Hence, each hit on one of the X, Y or Z keys will go to the next valid Xs, Ys or Zs.

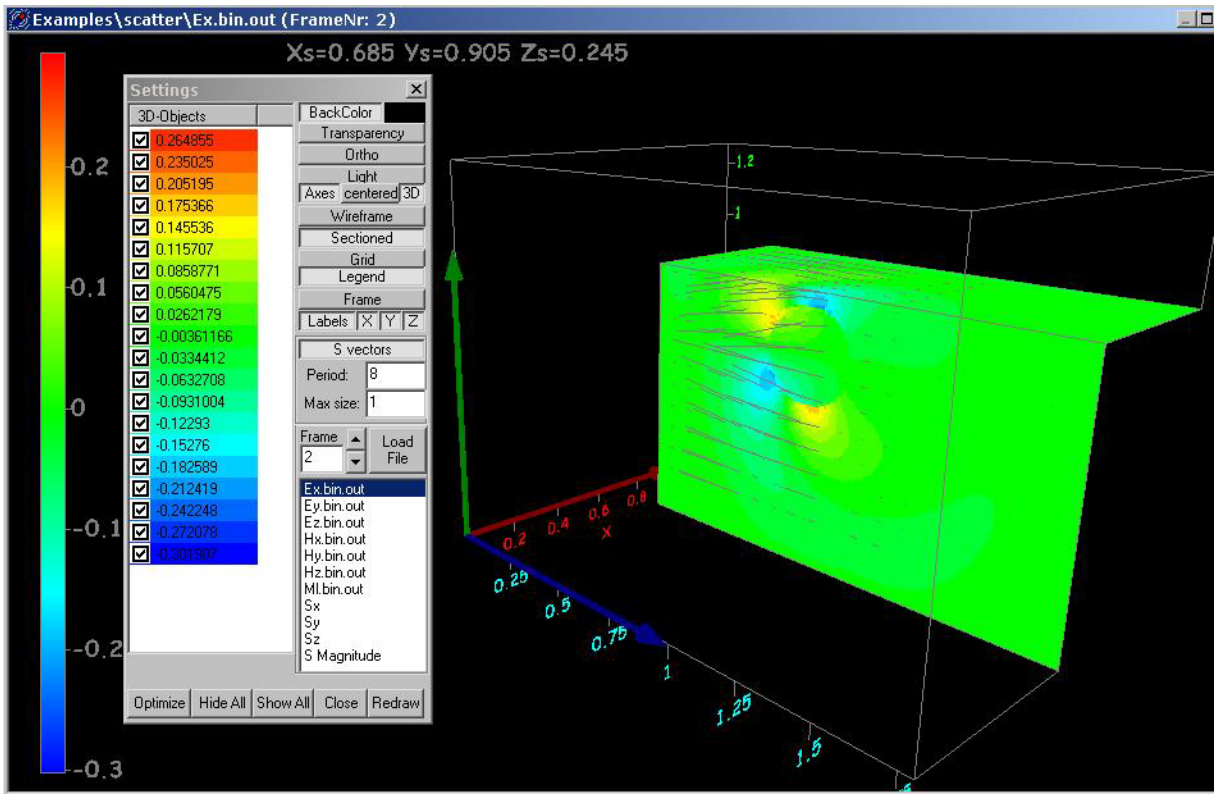
- **Grid** – will determine the drawing engine to paint a “cage” around the computational domain.

- **Legend** – enables or disables the drawing of the color legend on the right side of the screen.

- **Frame** – will draw a rectangular box around the computational domain. Notice that this “Frame” has nothing to do with the “Frame number” which is related to the exported field files.

- **Labels X,Y,Z** – enables or disables individually the drawing of the labels attached to each of the axes. The labels are displayed in the same basic color as the axis, which is red for x, green for y and blue for z.

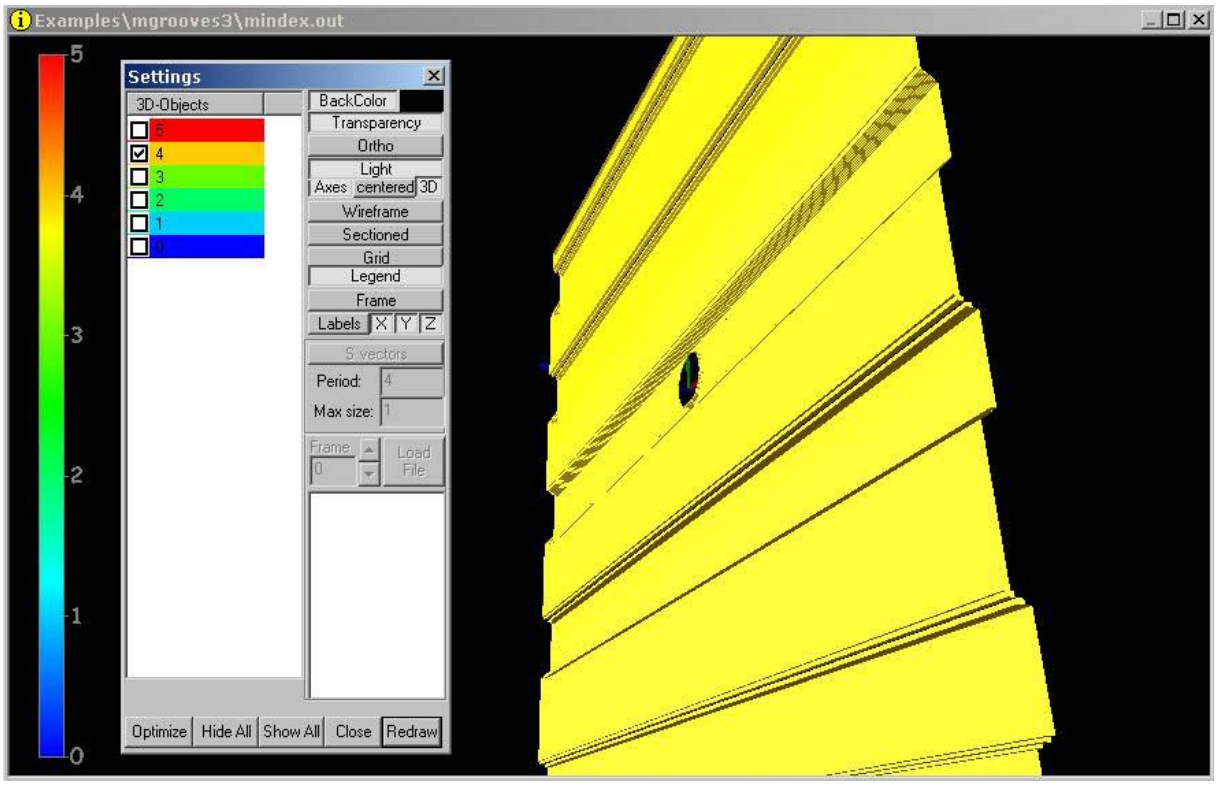
- **S-Vectors** – when the Solid Scene is displaying exported fields, with this option the computed S-vectors can be displayed, as indicated in the picture below.



This option takes effect only in conjunction with the ***Sectioned*** option, described above. Each S-vector is displayed as a grayed line having the foot-point located on the cross-section plane. To avoid a crowded picture, the arrow on the end of each vector is not drawn. For the same reason, a period (integer) number n may be specified, which means that the S-vector is computed and displayed only for each n-th point of the computational grid. (n is 8 in the above example). Finally, the ***Max size*** number is a simple scaling factor used only for displaying the length of the line that represent the S-vector with the greatest magnitude. The lengths of all other S-vectors will be adjusted proportionally.

- **Load File** – a click on this button will start the loading of data from the selected file. To use it, first select one of the available files (e.g. Ex.bin.out, Ey.bin.out, etc) from the list, select or type in the desired frame-number, then, finally, click on ***Load File*** to determine the graphic engine to load the data. Note that each of the files that contain one of the exported field components (e.g. Ex.bin.out, Ey.bin.out etc.) may contain more then one "snapshot" of the computational domain, each taken at different times. These "snapshots" (like frames in a movie) are stored sequentially in the same file and can be accessed directly by specifying the frame number. Only one frame can be viewed at a time.
Depending on the complexity of the data, the loading process may need several minutes until it is finished. The activity is displayed in the status bar of the main window, together with various other changing internal data, mainly to indicate that the process is going on.

- **3D-Objects list** – displays a list of all 3D-Objects that make up the scene. Furthermore, for each of the objects, it can be specified if it should be drawn as a wire-frame, or if it should be drawn at all. Initially, all 3D-Objects are visible. To remove some of them from the screen, simply uncheck the checkbox attached to it in the list. The following picture illustrates the displaying of only one of the 6 objects of the scene, namely the 3D-Object that has attached the value 4. (As a comparison, the full scene is displayed in the first picture of this chapter)



To enable/disable more then one object, just hold the mouse button pressed and move slowly the mouse cursor over the checkboxes. Each of them will toggle its state as the cursor enters the area of the checkbox.

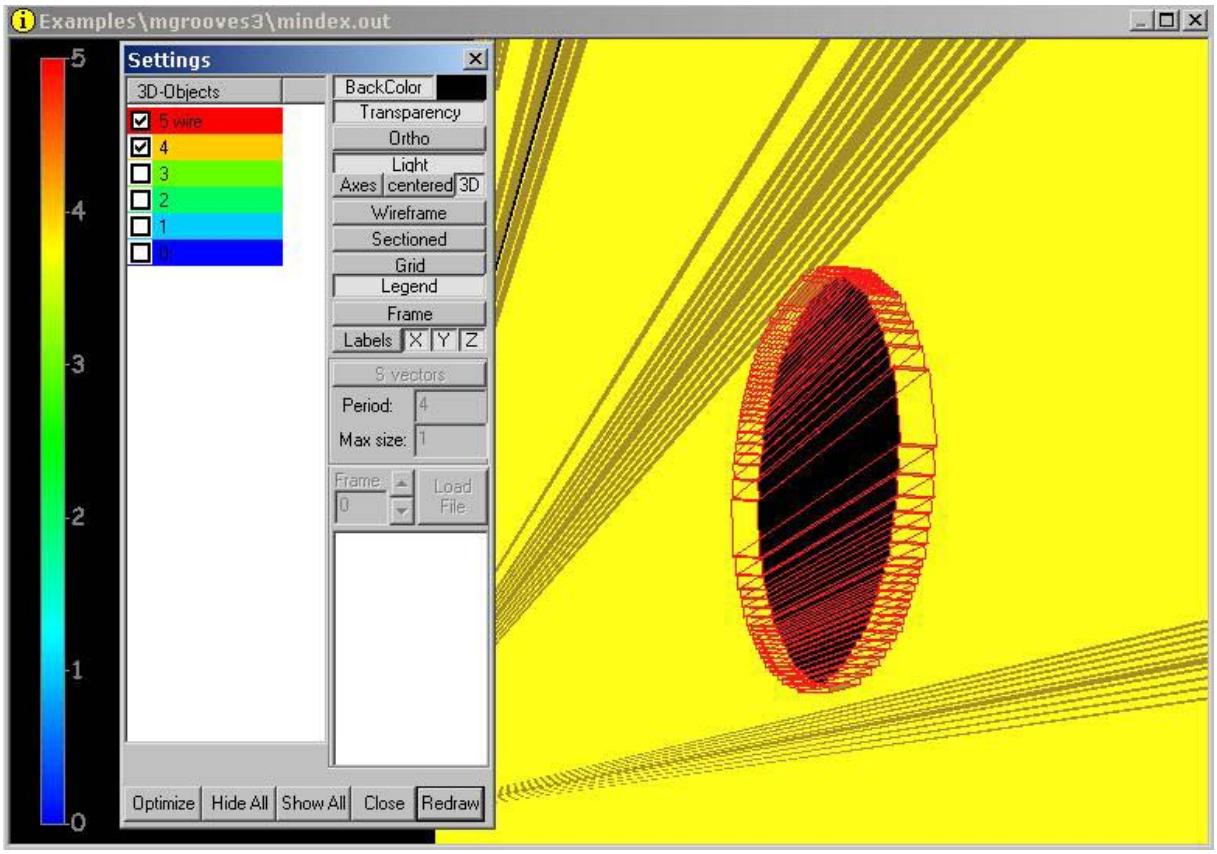
To show them all, click on the **Show All** button. This will check all available 3D-Objects in the list. In contrast, the **Hide All** button will uncheck all 3D-Objects, removing them from the scene.

After modifying the states of the checkboxes the changes will not be done into the scene until the **Redraw** button is clicked or the Enter key is pressed on the keyboard.

The **Optimize** button will perform the same algorithm as described in the section about loading and preparing the data. However, since the raw data is already loaded from the file, the re-loading is omitted and only the re-building of the 3D-Objects is initiated, but in a slight different manner. During the initial loading, the 3D-Object building is kept as fast as possible, with the risk of having undetected identical neighbored areas, which may lead to more primitive boxes then necessary. With the **Optimize** button, this process will ignore the speed, but will look more carefully to limit the number of boxes that made up an object.

Finally, the 3D-Objects list allows to specify for each of the objects if it shall be drawn as a wire-frame. To accomplish this, just click on the colored line entry of the list. As a response,

the keyword “wire” will be displayed together with the value attached to the object. The next picture enhances the last example by displaying the red object (value 5) as a wire-frame in addition to the already displayed yellow object (value 4), displayed as a solid object.

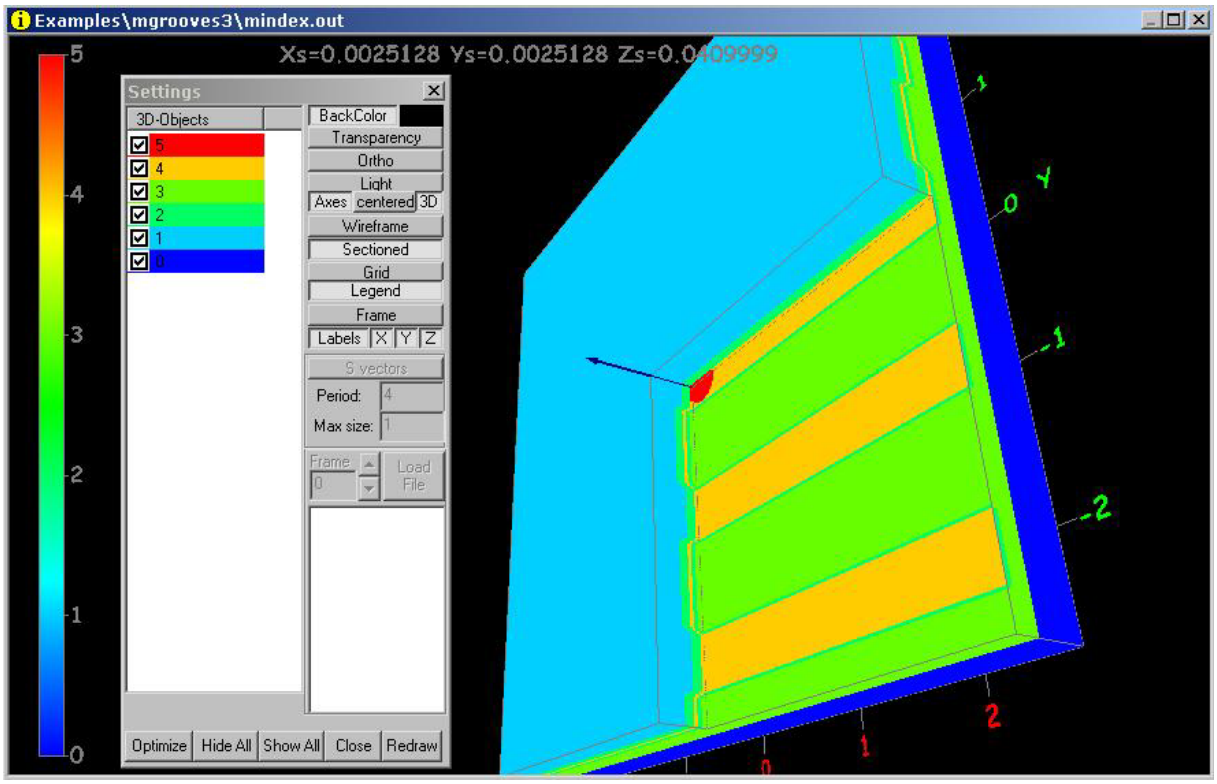


The example picture displayed on the next page illustrates the **Sectioned** option, applied to data loaded from a material index output file, as generated by Sim3D_Max™.

As a difference to the sectioned scene that displays field data, in this case not only the cross-section planes are displayed, but also the side walls of the geometry.

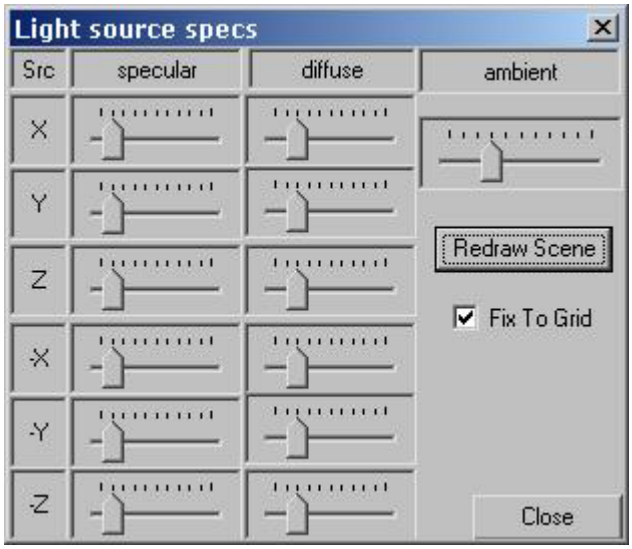
For best visualization, the **Transparency** and **Light** options should be disabled in this case.

The Xs, Ys and Zs coordinates of the section planes can be changed only with the X,Y and Z keys on the keyboard. In this mode, their values are displayed on top of the screen.



5.2.3. Lights

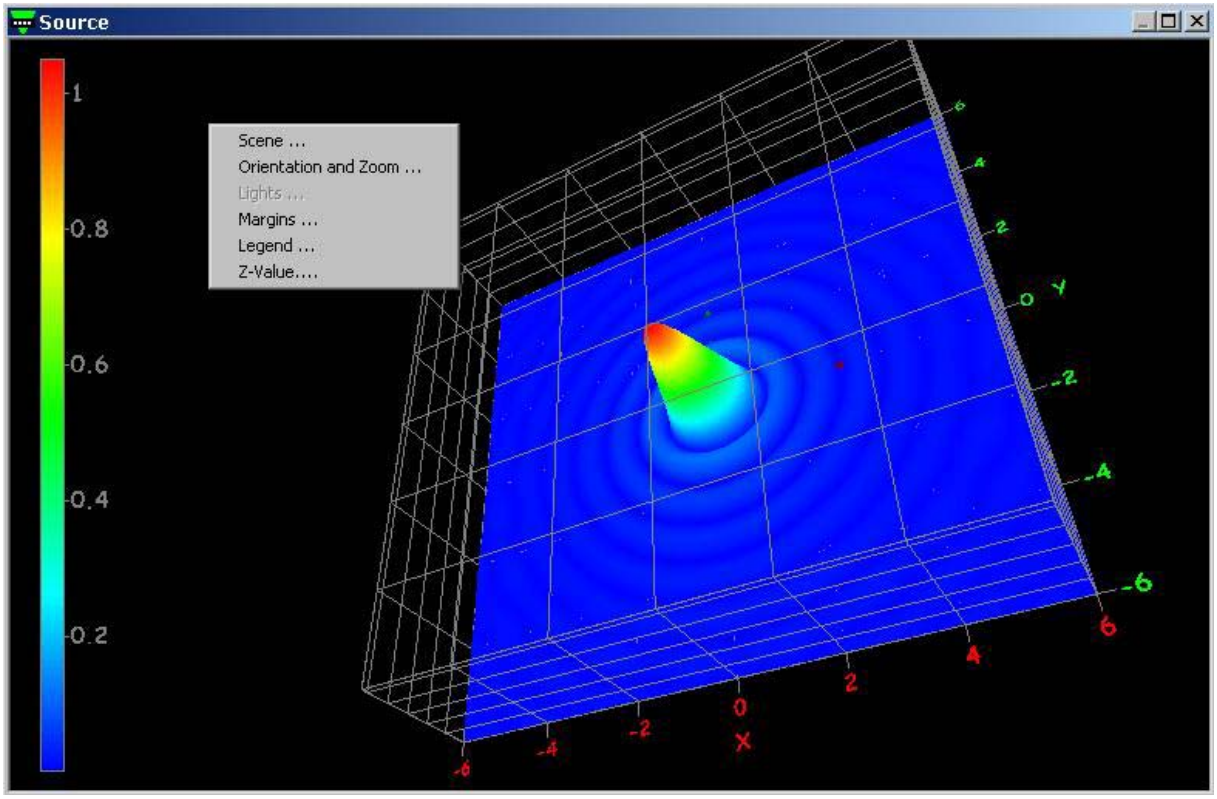
There are 6 internal light sources that are used by the framework to illuminate the scene. They are placed on the middle of the 6 sidewalls that make up the computational grid. The light emitted by them has a specular and a diffuse part; these intensities are used internally in different ways to compute the aspect of the scene. Additionally, for this purpose, an ambient intensity is considered, which is not attached to a specific source.



The properties of the emitted light of each source may be adjusted with the above dialog box, which is opened by the **Lights…** menu command of the popup menu. Since it will be seldom necessary to perform such modifications, this dialog box has been introduced for completeness only.

5.3. Relief Scene

In contrast to the Solid Scene, where tri-dimensional arrays are displayed, the ***Relief Scene*** mode is used to display two-dimensional arrays of complex data, like beam distributions and exported reflected/transmitted fields. Hence, this type of View3D will be automatically opened by the ***View 3D*** command of the popup menus attached to the ***Export reflected***, ***Export transmitted*** and ***DIFFRACTSource*** items of the *TreeView*.



As an example, the above picture shows the aspect of the Relief Scene as it appears immediately after loading a DIFFRACT source distribution.

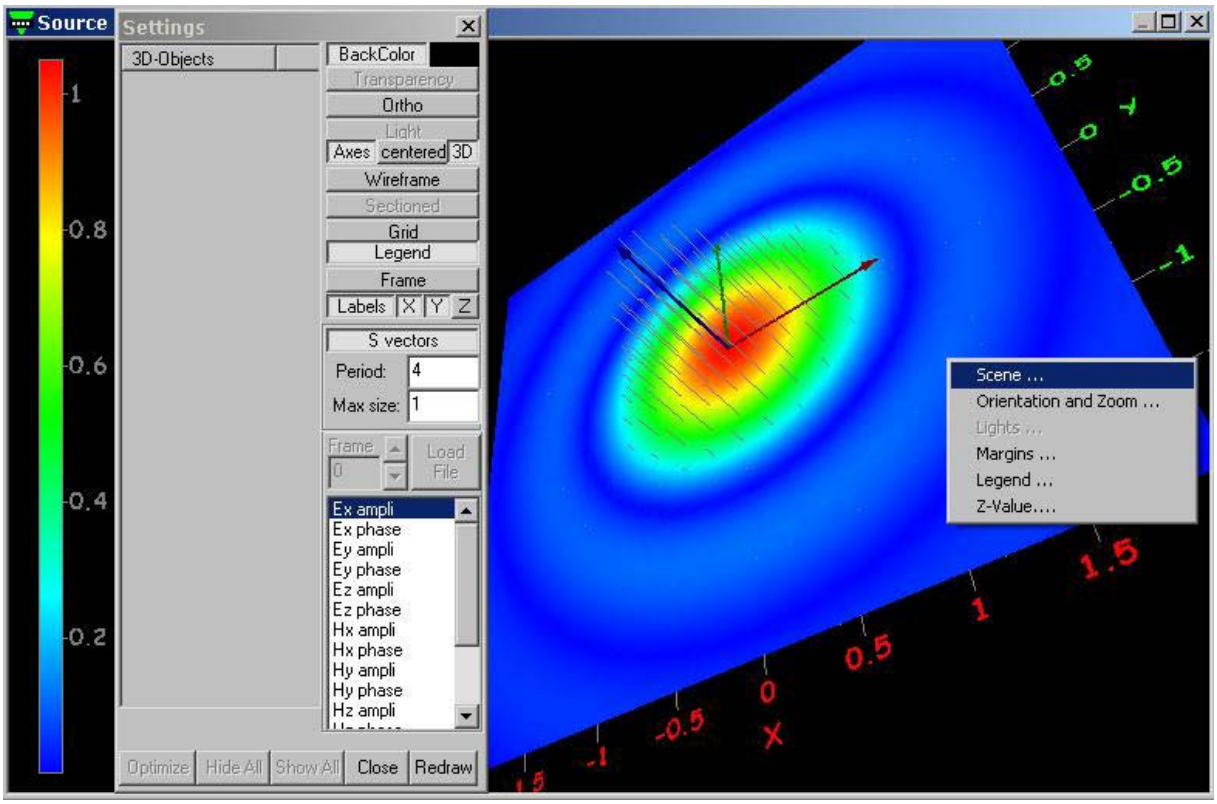
The orientation and zoom of the scene can be changed as described in the section dedicated to the common features of the View 3D.

The popup menu, which is displayed upon a right-button mouse click, contains several commands that will be described on the following pages.

5.3.1. Scene parameters

The scene options dialog box is invoked by the popup menu command ***Scene…*** It is similar to the scene options dialog box used for solid scenes, but some of the controls that are not used here are grayed out. For example, there are no 3D-Objects involved in the ReliefView, hence the *transparency*, *light* and *sectioned* options are disabled and meaningless. Also, there is no need to reload any file, since all the arrays are loaded once and kept in the working memory, so the *Load File* button is disabled too.

A slight difference may be noticed when the ***S vectors*** option is selected. In this case, the data matrix is displayed flat at z=0, without any elevation, and the lines that represent the S-vectors (which are computed similarly to the S vectors of the solid scene) start all in the z=0 plane. Again, for simplicity, as in the case of the solid scene, the arrow of the vectors is not drawn, as illustrated in the picture below.



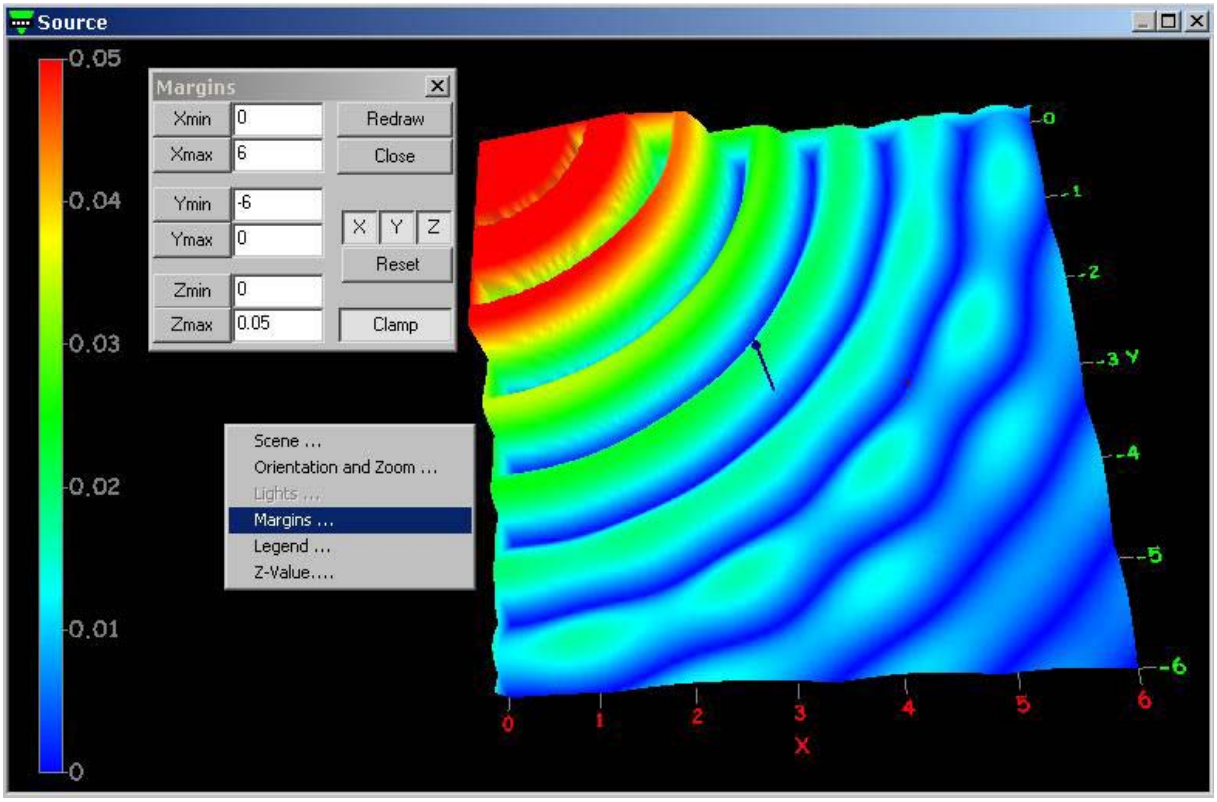
As mentioned, the data files that serve as input for the *Relief Scene* contain multiple two-dimensional complex arrays, one for each field component. Each of the complex arrays are divided in two real arrays, one for the amplitude and one for the phase. Any of the available arrays, listed in the Scene options dialog box, can be selected into the scene (one at a time). In the above example it is the *Ex amplitude* data displayed.

The other control elements work similar to the solid scene. Please refer to that section for more information.

5.3.2. Margins

The extents of the three-dimensional grid that confines the displayed mesh are usually taken from the loaded input file and will cover the whole data area. However, for a better visualization of small details on the mesh, these extents can be modified with the ***Margins*** dialog box. It is invoked by the ***Margins…*** command of the popup dialog box.

As an example, the following picture shows the effect of this feature, applied to the same data as displayed in the first example picture of this section.



To modify the margins, type in new values for Xmin, Xmax, Ymin, Ymax, Zmin and Zmax and click on the **Redraw** button.

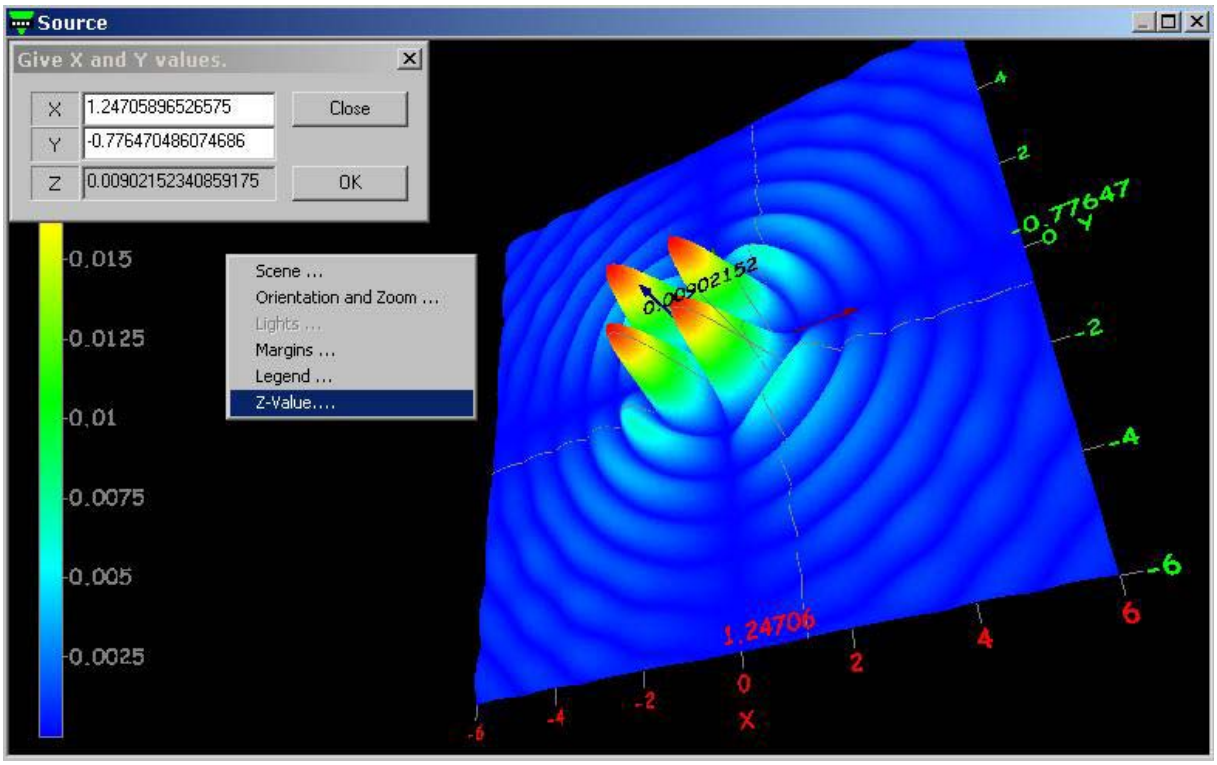
To restore back the various min-max value pairs, select or deselect the desired X,Y or Z buttons located above the **Reset** button and click on the **Reset** button.

The **Clamp** option will truncate the mesh that lie outside the specified margins, making the display more readable.

Finally, it should be noted that the tree-dimensional grid size is not modified by changing its margins. Moreover, it is the mesh data that will be displayed differently. As such, distortions are likely to appear, if the width-depth-height ratio factors of the new margins are not the same as for the old ones.

5.3.3. Z-Value picker

The Z-Value picker is a small dialog box, invoked by the ***Z-Value…*** command of the popup menu, which allows to find out the Z-value at given (X,Y) coordinate pairs.



To use it simply specify the desired X and Y values and click on ***OK*** (or press the Enter key on the keyboard). As result, first, the X and Y values will be adjusted to the nearest coordinates for which a Z-value is available, then all three values (adjusted X,Y and retrieved Z) are displayed in the dialog box and (less accurate) in the scene, as indicated in the picture above. Furthermore, two cross-section lines intersecting at (X,Y) will be added to the scene.

A more convenient and faster method to retrieve the Z-value at any position inside the mesh is to use the mouse. Simply place the mouse cursor on the desired position and press the *middle* button of the mouse (usually this is the scroll wheel of the mouse, which has to be pressed and not rotated). Alternatively, with the same effect, hold the *Shift* key on the keyboard down and press the left mouse button.

Finally, to remove the cross-section lines and the X,Y and Z values from the screen, position the mouse cursor outside the mesh and click the middle button of the mouse (or hold down *Shift* and click the left button of the mouse).

6. Files and Directories

Many command blocks inside the *parameters.input* file refer to external files placed on the hard disk. Both programs, Sim3D_Max and Sim3D_MaxGUI must be able to find and access those files for proper operation. This is of course not a problem if the filenames are given with fully qualified pathnames (like *C:\SomeFolder\AnotherFolder\filename.ext*). But if the filenames are given only in the short form without any path information (like *filename.ext*), as it is usually the case, some additional information is needed.

This chapter describes how Sim3D_MaxGUI builds up the required fully qualified filenames from the short filenames that are usually specified in the *parameters.input* file.

The most important additional information needed is the ***Working Directory*** entry in the *parameters.input* file and the location folder of the Sim3D_Max executable. See next chapter about how to set the Sim3D_Max location folder.

Following rules are used:

- if the filename is given as simple name (like *filename.ext*) or with a relative path (like *\Examples\filename.ext*), then the fully qualified name is build up by concatenating the Working Directory with the given filename.

- if the file is specified with a fully qualified name (like *C:\SomeFolder\AnotherFolder\filename.ext*) then no other modifications will be done.

- if the Working Directory is specified as a relative path (like *\Examples\bowtie*), then it will be always considered as relative to the location folder of the Sim3D_Max executable.

To accomplish the last rule, Sim3D_MaxGUI always sets the current directory of the file system as the location folder of the Sim3D_Max executable.

When specifying a relative path, never forget the beginning backslash \.

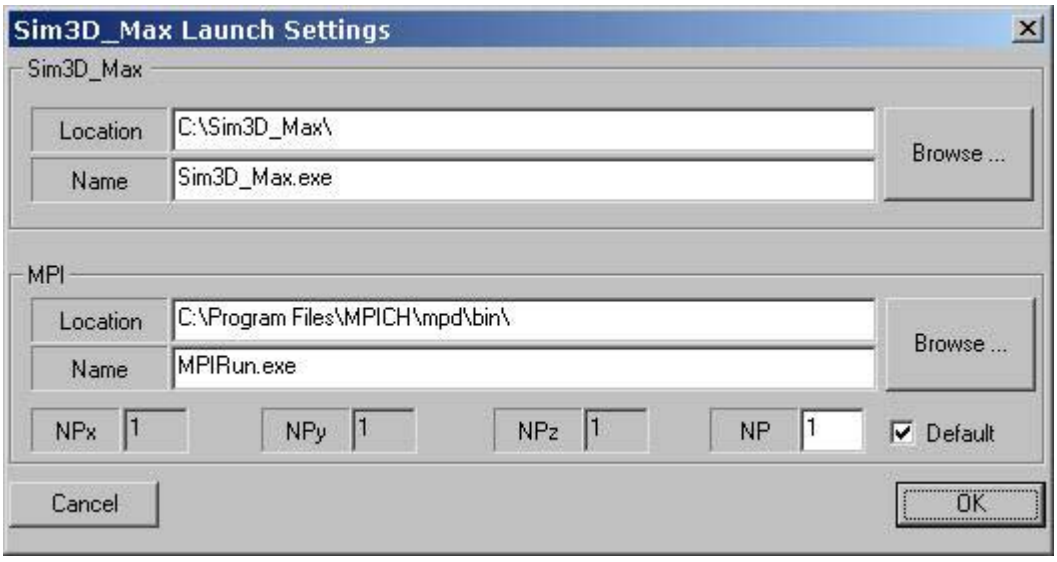
7. Launching Sim3D_Max

After finishing the preparation of the *parameters.input* file with Sim3D_MaxGUI, the computing session with Sim3D_Max can be started with the menu command ***Sim3D_Max – Launch!***, or ***Sim3D_Max – Launch MPI!*** as figured out in the following picture:

Simultaneously, the actual loaded *parameters.input* will be passed to Sim3D_Max for further processing.

As a quick shortcut, instead of the menu commands, the toolbar buttons or may be used, with the same result.

To be able to start Sim3D_Max, Sim3D_MaxGUI needs to know how the Sim3D_Max - executable is named and where it is located. You must specify this with the **Launch settings …** menu command. The following dialog box appears:



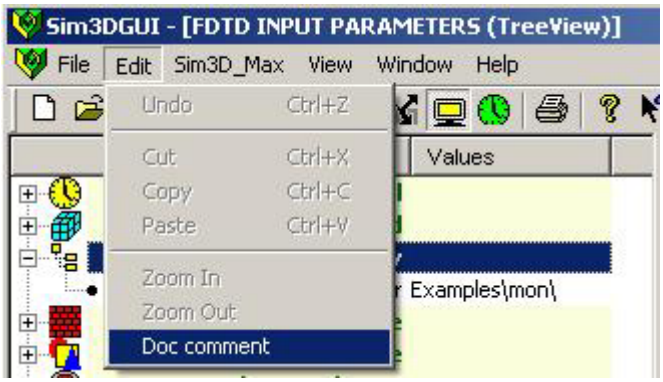
The name and location of Sim3D_Max may be typed in directly (do not forget the extension **.exe** and the terminating backslash \) or, more conveniently, they may be searched for with the **Browse…** button. Similarly, the name, location and calling parameters of the MPI executable can be specified directly or with help of the **Browse …** button. This information will be used when the ***LaunchMPI*** command is invoked (please read the Sim3D_Max User Guide for further information)

8. Three Words

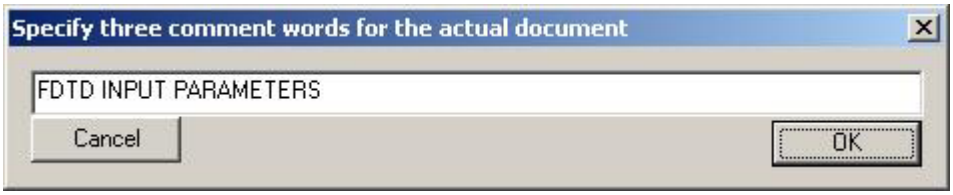
According to the Sim3D_Max manual, each *parameters.input* file must begin with three comment words, separated by spaces. Sim3D_MaxGUI uses this as a preliminary check, to decide if a file requested to be opened, is a *parameters.input* file or not.

Furthermore, the three words will be displayed always in the caption text of the main windows of Sim3D_MaxGUI.

These words may be changed anytime with the menu command Edit – Doc comment, as illustrated below:



A small dialog box will be opened, allowing to type in the desired comment words:



If the comment text has more than three words, then the text will be truncated. If it contains less than three words, then the missing words will be replaced by question marks.